

SurgeMail PDF document Index

This document is periodically autogenerated from our online documentation. As such there is no real structure to the pages in this pdf. You probably want to read it more like website by following key links rather than as a linear book.

For the latest information you should see the online: [surgemail help](#) and [surgeweb help](#)

Some key locations within this document to know about:

1. [SurgeMail help index](#) covering surgemail mailserver basics and configuration guides
2. [SurgeMail change history](#) Core surgemail change history. For a more detailed listing of all builds as they are released see the [SurgeWeb change history](#)
3. [SurgeWeb help index](#) Documentation of the modern Web 2.0 surgeweb webmail interface included as part of surgemail
4. [SurgeWeb change history](#) Documentation of surgeweb changes, and history of all surgemail builds as they get made available as prerelease specials builds.

SurgeMail Help Index

[SurgeMail Home](#) | [Download](#) | [Latest Manual Online](#) | [Manual in pdf format](#)

1. [SurgeMail in a nutshell](#)
2. [Getting Started](#)
 1. [Before you install, what you need to do.](#)
 2. [Installation and Upgrading](#)
 3. [Migration to SurgeMail](#)
 4. [Change history](#)
 5. [Post Installation Instructions](#)
 6. [FAQ](#) Frequently Asked Questions
 7. [Customer Support](#)
3. [Spam and Virus Protection](#)
 1. [Virus Protection](#)
 2. [Spam Prevention](#)
 3. [Mail Filtering](#)
 4. [Friendly Relations System](#)
 5. [Realtime Blackhole Lists \(RBL's\)](#)
4. [Managing your Mail Server](#)
 1. [Server Status](#) information
 2. [Searching the log files](#) (tracking a message)
 3. [Report](#) generation
 4. [Managing accounts](#)
 5. [Using the tellmail utility.](#)
5. [SurgeMail Configuration Settings](#) ([overview](#))
 1. [Domain specific settings](#)
 2. [Global settings](#)
 3. [WebMail settings](#)
6. [Configuration Guides](#)
 1. [Authentication Modules](#)
 2. [Virtual Domains](#)
 3. [Clustering](#)
 4. [Mirror the server](#)
 5. [Performance and Scalability](#)
 6. Customising [look and feel](#)
 7. Customising [internal emails](#)
 8. [Language translation](#)
 9. [Mail Redirection](#) (forwarding, aliasing, gateways)
 10. [SurgeWall](#) - SurgeMail filtering on existing mail servers!
 11. [SMS](#) - various SMS delivery options
 12. [Mailing lists and bulletins](#)
 13. [Securing the server](#)
 14. [WebDav](#)
 15. [CalDav](#)
 16. [Incoming MX servers](#)
 17. [Smart Router / Load balancing](#)
 18. [NDB NetWin folder format](#)
 19. [DomainKeys Support](#)
 20. [IPV6 Support](#)
 21. [SurgeVault Encryption feature](#)
 22. [Example configurations](#)

SurgeWeb Help Index

This is the surgeweb online documentation index.

- 1. Overview
 - 1. [Introduction and key product benefits](#)
 - 2. [Surgeweb FAQ](#)
 - 3. [Surgeweb Customisation](#)
 - 4. [Surgeweb Contacts handling](#)
 - 5. [Labels, Searching and spam handling](#)
- 2. Installation and latest versions
 - 1. [Change history](#)
 - 2. [Known bugs and feature requests](#)
- 3. Advanced configuration guides
 - 1. [Multilanguage support](#)
 - 2. [Surgeweb Spam handling if upgrading](#)
 - 3. [Clustering configurations](#)
 - 4. [Performance](#)

SurgeMail Change History

SurgeMail 6.0 Features

Companion Email Client for Window,Mac,IPhone, Ipad released: [YesImOnline email client beta](#)

Quota system improved to recover better from unclean shutdowns

Imap settings added:

```
imap_public_show "true"
  Auto subscribe to public folders
g_imap_unsub_auto "true"
  Auto unsubscribe from deleted folders
g_sent_store "System_Sent"
  Put all sent messages in the named folder.
```

Other settings added:

SECURITY_SUFFIX - Domain based setting to stop hackers guessing accounts, this requires everyone to login with a different suffix from their actual domain name, e.g. user@xyz.com would login as user@secret.xyz.com

SurgeWeb - Summary

- Multiple Account support!
- Access to Legal Archive
- Multipart related image support in the signatures to support "business card" type images as part of the signature.
- Emoticon icon support as multipart related inline images when composing a message
- Ability to add multipart related inline images when composing messages. Includes serverside rescale and crop facility as part of the upload dialog.
- Signature improvements - now supports multiple signatures, of unlimited size, with the ability to set the default for each additional account.
- Full details of SurgeWeb [changes](#).

Misc Bug Fixes including...

```
html_bounds bug fix.
Bug fix for chrome editing blog posts
Fixed truncation of ipv6 headers - 6.0b-22
```

SurgeMail 5.0 New Features

SurgeMail 5.3h

- New Spam handling see <http://netwinsite.com/surgemail/help/myrbl.htm>
- New Legal Archive - http://netwinsite.com/surgemail/help/legal_archive.htm
- SurgeWeb - New Improved Web Email Interface - [changes](#)
- New settings to prevent/reduce harvesters and hackers, see g_hacker* settings.
- New basic support for Amazon SES - <http://netwinsite.com/surgemail/help/amazon-ses.htm>
- New Basic IPV6 support added
- Imap public folders improved.
- SurgeVault - [See here for details](#)
- WebDav support - [See here](#)
- New [G_FRIENDS_BOUNCE_FRIEND](#) "true" - This makes friends 'bounce/reject' exceptions work even if the person is a friend.
- New [g_inbox_max](#) "int" - This setting will stop users leaving lots of message

in their inbox

- New [G_INBOX_NOLIMIT](#) "int" - Use for special users who are not subject to the normal message count limit on their inbox ([g_inbox_max](#))
- New [g_outgoing_n](#) "int" - New settings to identify outgoing spam and email the manager if a limit is exceeded for any particular user per day.
- Blacklist ip addresses and block incoming email based on train/spamtrap hits, see: [g_black_isspam](#)
- New captcha to bounce messages.
- FIX: RBL timeouts are now more reliable with multiple lookups.
- FIX: ASPAM updates now more reliable on certain networks (failing on some).
- FIX: [vanish_bad_bounces](#) - broken in version 5
- FIX: Quota drift issue - deleted messages weren't being removed from quota in some circumstances.
- Fix: Memory Leaks
- Fix: Various crashes and improved performance.
- Fix: Issues with friends confirmation emails.
- Fix: [redirect_cc](#) - Failed with comma separated lists.
- Fix: File handle leaks.
- Fix: imap handling where wrong message body could be delivered to user.
- Fix: imap issue that caused problems with outlook
- Fix: Timezone issues
- Fix: imap/pop lockups in some circumstances.
- Fix: [g_fix_crcrlf](#) "bool" - caused corruption of messages

SurgeMail features of possible general interest added in recent months (as of 4.2g-26, February 2010):

- Blacklist ip addresses and block incoming email based on train/spamtrap hits, see: [g_black_isspam](#)
- New settings to prevent/reduce harvesters and hackers, see [g_hacker*](#) settings.
- New setting to block well known hackers/harvester, see [g_honeypot*](#) settings (not recommended)
- Prevent local address harvesting, [g_dlist_nolocal](#) true
- Basic IPV6 support added
- Manager warning when smtp thread count exceeded ([g_smtp_warning](#))
- Disable smtp authentication for non trusted addresses see: [g_smtp_auth_off](#), [g_smtp_auth_ip](#) or [g_relay_allow_ip](#)
- Block outgoing email to known phishing addresses (dubious value) [g_spam_phishing](#) "true"
- Setting to reject any email with surbl web content (spam urls) [g_surbl_reject](#) bool (reduces spam but will cause some false positives)
- Check whois info on suspect urls (this is unwise for busy servers as it may get your server blacklisted with the whois servers) A cache is used to minimize the load. ([g_surbl_reject](#) true) (**Requires 4.2g-27**, prior to that not stable!)
- Imap public folders improved.
- SurgeVault - [See here for details](#)
- WebDav support - [See here](#)
- [g_safe_smtp](#) (requires 4.3f-20 or later) this setting prevents smtp authentication unless the net block has had a successful imap/pop login (this prevents most hackers sending spam via your server even after they guess a valid password)

Version not yet released but being currently worked on :

- New: Lots of [surgeweb features](#), and various fixes and other new features. A new full beta build will be done soon, and notes documenting all changes will be added soon.
- New: [g_bull_rule](#) new controls for bulletin access.
- New: [g_breakin_white](#) - new spammer detection.
- New: [G SMTP PORTFORCE](#) - Enforce smtp authentication 'only' be used on the ports specified.
- New: [g_redirect_cc_attach](#) - this feature lets you send messages 'exactly as they look on delivery to a user and only the messages that get delivered, to another account, as mime attachments.
- New: updated SSL version used in Windows.
- New: Added ownership to archives, The owner gets an 'archives' button in their user self admin page.
- New: Added honeypot support <http://www.projecthoneypot.org>
- New: Improved dlist speed with large messages, 10-30 times faster now.
- New: Support for ip checks in any ip setting for ipv6 addresses and fixed several bugs with ipv6.

- Fix: error that causes bounces to be sent still in the backscatter code.
- Fix: many bugs and optimisations.
- Fix: not handling quotes properly in an email address.
- Fix: file handle leak.
- Fix: an issue with case checks on vdomains.
- Fix: issue with imap code deleting wrong message in rare occasions.
- Fix: NDB on 64bit linux and also with target bucket files over 2 gigs also some other fixes with the NDB code that could cause crashes.
- Fix: Several bugs that could cause crashes.
- Fix: [g_authent_always](#) - was completely broken.
- Fix: Issues with mirroring and surgevault and a minor issue with mirror flags.
- Fix: Mirroring issue with responders.
- Fix: Issue with blog permissions.

SurgeMail 4.2d4-4 17-February-2010 (Webmail 3.1t-12)

- Fix: Patch of current production release for crashing bug for certain oddly formatted spam messages.
- Fix: Fixed missing files in the help pages installed on the mailserver.

SurgeMail 4.2d3-3 28-January-2010 (Webmail 3.1t-12)

- Fix: Get the "download all fix" noted in [surgeweb changes](#) into the production release distributions. Possible signs of this issue having been hit: Major jumps in diskusage by the surgeweb folder, restart reports in startstop.log referring to MZIP mutex. Note: Subsequent to this build (4.2d3-3) it was noted that if this issue is hit, these temp folders files are not actually getting deleted after a few days as I thought they were, this is fixed in builds 4.2g-6+ (or if this space needs to be recovered more urgently manually delete large attachments.zip files in the surgemail/surgeweb/work tree).

SurgeMail 4.2d2-2 5-December-2009 (Webmail 3.1t-12)

- Fix: Several surgeweb fixes already in the 42e* builds

SurgeMail 4.2d-1 5-November-2009 (Webmail 3.1t-12)

- New: SurgeWeb changes as per: [surgeweb changes](#)
- Fix: Prevent invalid pstat entries that were getting created on some systems and fix the current pstat.dat database (some systems this was HUGE)
- Fix: Windows only system library timezone bugfix to fix the off by one hour issue
- Fix: Fixed occasional unreliability of nwauth mirroring (required both mirrored servers to be upgraded)
- Fix: Bug if g_friends_add_trusted used
- Fix: Several sporadic crashing bugs

SurgeMail 4.2a3-3 9-October-2009 (Webmail 3.1t-12)

- Fix: Doing a surgeweb "download all" of multiple attachments with no file extension and the same name crashed surgemail
- Fix: Surgeweb forwarding a message with attachments or forward attach was losing the attachment at send time if using the basic html interface
- Fix: Friends processing bugfix

SurgeMail 4.2a2-2 24-September-2009 (Webmail 3.1t-12)

- New: Extensive improvements to the SurgeWeb "Ajax / Web 2.0" web email interface (for more information see [surgeweb changes](#), and [known bug list](#))
- New: SurgeVault email encryption (feature is PRERELEASE - contact surgemail-support@netwinsite.com if you want to try using this)
- New: Force use of ssl on webmail/surgeweb/user.cgi (but not blogs and surgeplus) using g_ssl_require_web
- New: Improved DNS handling sending requests to multiple servers
- New: Split g_from_valid into g_from_valid (recommended) and g_to_valid (unwise to use)
- New: Allow use of spaces in passwords (NOT RECOMMENDED) using g_authent_spaces and nwauth 4.0r+
- New: To header added to the HTML spam status email

- Fix: Domain quota issue (not workign if mailbox had trailing slash)
- Fix: Naked LF could slip through in very specific circumstances
- Fix: Some imap response fixes to make sure thunderbird did not get confused sometimes wrt showing new messages
- Fix: Retry times were not getting obeyed for messages that could not even open a connection to the destination server
- Fix: Webmail trash emptying related quota drift issue
- Fix: Two mutex related fixes
- Fix: Some imap response fixes to make sure thunderbird did not get confused sometimes wrt showing new messages
- Fix: Minor IPV6 fixes

SurgeMail 4.0v-8 3-June-2009 (Webmail 3.1t-12)

- New: Surgeweb searching revamped - works across multiple folders, and messages have "search by subject" (thread) / "search by address" conversation history with sender. For more info see [surgeweb changes](#)
- New: HTML "spam status" message (status_html.eml) that allows friends pending spam store actions to be done from normal mail client. Actions allowed: delete message; deliver message; purge spam store; show log. To revert old behaviour use g_friends_old_status_email
- New: Linux builds now support "Load factor" and "CPU utilisation" in trend graphs

SurgeMail 4.0u4 10-June-2009 (Webmail 3.1t-12)

- Fix: memory leak fixed in probe code
- Fix: webmail has correct version number
- Fix: domain with letters a-f only fix is actually in this build

SurgeMail 4.0u3 28-May-2009 (Webmail 3.1t-11)

- New: Continued improvements to the SurgeWeb "Ajax / Web2" web email interface (for more information see [surgeweb documentation](#), [surgeweb changes](#), and [known bug list](#))
- New: Blogs make use of surgeweb cross browser html editor - old editor / text mode can be enabled per blog in advanced settings
- New: Basic support for IPV6 under windows and linux (beta and requires g_ipv6_enable)
- Fix: webmail template fix so that the html editor is not displayed for IE8 (the same as has always been done for IE7). The fix is part of this build - the webmail version was upped to 3.1t-12, but the incremented version number did not make it into the build :-)

SurgeMail 4.0k 10-April-2009 (Webmail 3.1t-11)

- New: Continued improvements to the SurgeWeb "Ajax / Web2" web email interface (for more information see [surgeweb documentation](#), [surgeweb changes](#), and [known bug list](#))
- New: User.cgi has a blocklist feature (just like friends but blocks by address) so you can block by address without having to add a zillion filtering rules
- Fix: Trend graphing overflow problem + several fixes to several issues that became apparent as result of first fix
- Fix: Mirroring of settings that don't exist on the master
- Fix: CPU bug if ports were disabled
- Fix: IMAP idle command was not showing new messages under certain conditions for non inbox folders
- Fix: Timezone related "out by one hour" fix for windows 2003
- Fix: Several fixes dlist member removal via web admin interface
- Fix: Several fixes to surgeplus photo sharing
- Fix: Crash when some messages (generally spam) were viewed through user.cgi
- Fix: Archive viewer was missing the last 30 odd characters of messagas
- Fix: Forward / responder was failing to respond and deliver locally in some circumstances
- Fix: Mutex locking problems on multicpu servers
- Fix: Made g_imap_timeout apply to the idle command
- Fix: Dlist memory leak
- Fix: Lockup if authent module doesn't respond nicely to -version request

SurgeMail 4.0a 5-January-2009 (Webmail 3.1t-10)

- New: SurgeWeb new "Ajax / Web2" high performance web email interface (still being worked on though, for more information see [surgeweb documentation](#), [surgeweb changes](#), and [known bug list](#))
- New: New archive searching interface with CRC validation to confirm the messages in the archive have not been tampered with
- New: Arbitrary error translation of error messages (g_error_xlate)
- New: Setting to adjust max number of messages in a folder (g_imap_max_messages) - defaults to 200,000
- New: Imap burst login to delivery log so message->uid renames can be traced
- New: Default page not returned for invalid cgi requests - stops "dumb tools" reporting surgemail as compromised (old behaviour can be restored with g_web_old_behaviour)
- New: Bounce other messages if first message is mending awaiting friends bounce (g_friends_bounce_second)
- New: Require friend confirmation if email appears to be in language not in list of accepted languages
- Fix: High use reports were getting confused by addresses longer than 100 characters (these get truncated in the delivery logs)
- Fix: Thunderbird timeout with large imap folders
- Fix: Tellmail archive search fixes for: g_maildir_netwin, date based archive before first rollover
- Fix: Sporadic crash in archive extract
- Fix: SPF related fix where DNS errors sometimes generated
- Fix: Message attachments sometimes not showing in Thunderbird
- Fix: CR/LF handling when attaching footer files on unix
- Fix: In FF3 the admin interface login resulted in multiple login dialog boxes
- Fix: Editing dlist/lists.dat directly in raw format from admin interface sometimes deleted it completely
- Fix: Responder now works when late forwarding is used
- Fix: Webmail APOP was sporadically crashing webmail
- Fix: Avast for windows would stop messages with very high compression ratio with an error, these are now allowed and logged with a "possible compression bomb" warning
- Fix: Address was being synched when mirroring config files, address field is now not mirrored
- Fix: Improvements to the noforgeme handling
- Fix: Bounce handling of surgewall filtering rules
- Fix: Admin login sporadically "froze" due to file lock on users.lst file
- Fix: A number of other minor fixes

SurgeMail 3.9g2 18-June-2008 (Webmail 3.1t-7)

- Security: Minor issue allowing imap command to crash surgemail

SurgeMail 3.9g 13-June-2008 (Webmail 3.1t-7)

- New: Allow headers "exists" check in users filtering rules
- New: Allow raw message content to be displayed from user cgi spam and friends pages
- Fix: User filtering "or rules" were not correctly handled sometimes.
- Fix: Imap login on mirror server was not deleting old messages
- Fix: Originating ip and orbs headers not added for authenticated users
- Fix: Various other various minor fixes

SurgeMail 3.9e 10-April-2008 (Webmail 3.1t-7)

- Fix: Minor new IMAP module fixes
- Fix: Several other minor fixes

SurgeMail 3.9c 14-March-2008 (Webmail 3.1t-7)

- Fix: pstat database file backed up and cleared when switching binaries with incompatible data types (eg. switching 32 / 64 bit binaries) - previously this would corrupt pstat database
- Fix: Blacklist related crash fixed
- Fix: Improved quota handling in new imap code
- Fix: Reduced disk loading on webmail (due to reduced flushing)
- Fix: Variety of other minor fixes

SurgeMail 3.9a 31-January-2008 (Webmail 3.1t-5)

- Fix: Linux 64 bit webmail issue (introduced Nov 2007)
- Fix: New IMAP fix

SurgeMail 3.8u 25-January-2008 (Webmail 3.1t-4)

- New: New setting g_spam_from_blacklist
- Fix: OSX mail client not handling some imap responses nicely
- Fix: Improved UIDL handling in new IMAP implementation
- Fix: Correctly keep imap flags during migration, this was broken by new imap implementation.
- Fix: Corrected html footer handling under some circumstances
- Fix: Corrected surgemail and webmail handlig of "%" when listing imap folders. NOTE: If you upgrade the surgemail binary and not webmail binary, users mail folders will seem to disappear from webmail. To correct this add "imap_list_* true" or make sure both webmail is upgraded with surgemail.

SurgeMail 3.8s 7-January-2008 (Webmail 3.1t-1)

- New: IMAP folder based access to friends pending (g_imap_friends)
- New: IMAP folder based access to aspm training folders (g_spam_folders, g_spam_folders_show)
- New: New spam prevention settings: g_from_valid, g_bounce_suggest, g_from_noforge, g_from_noforgeme, g_max_bad_ip, g_max_bad_ip_time, g_msg_max_drop.
- New: Miscellaneous new settings: gateway_to, old_pophost_bind, g_bind_incoming, g_friends_short, g_friends_ignore_trusted, g_friends_url, g_log_dropped_disable, g_rcpt_trace, g_redirect_ignore_errors, g_retry_unwarn, g_send_helo_in, g_spf_web_url, g_ssl_ciphers, g_ssl_disable_sslv2, g_virus_disable_local, g_web_noserver.
- Fix: Several fixes to the new IMAP implementation
- Fix: Webmail smooth template enabled for safari 2.0.4+ (Was already mostly operational for safari but several minor changes made)
- Fix: user.cgi editing last filter in filter list would delete it rather than save it
- Fix: Variety of other fixes and minor performance enhancements

SurgeMail 3.8q 26-November-2007 (Webmail 3.1s-18)

- New: Rewritten more efficient IMAP implementation
- Fix: Webmail' clickable links would sometimes be broken / display missing chunks from message
- Fix: Web Friends & account rename was partly broken for non primary domains
- Fix: Blogs bug that allowed two blogs of same name except for trailing slash
- Fix: Blogs postings with '<' or '>' characters is now handled nicely
- Fix: Webmail message list confusion when fetching mail from servers with with case sensitive POP UIDs (eg MS Exchange)
- Fix: Admin interface log searching would sometimes return an incorrect empty result.

SurgeMail 3.8o 30-August-2007 (Webmail 3.1s-12)

- Fix: Get packet fix into latest release too

SurgeMail 3.8k4 29-August-2007 (Webmail 3.1s-6)

- Fix: Improved version of the TCP packets small in relation to SMTP headers fix in 3.8k3

SurgeMail 3.8k3 28-August-2007 (Webmail 3.1s-6)

- Fix: Rather nasty bug that could corrupt message text if message headers were sent 1 TCP packet at a time.

SurgeMail 3.8m 20-August-2007 (Webmail 3.1s-11)

- Security: IMAP buffer overrun (affecting logged valid email accounts only)
- Fix: Variety of minor surgemail and webmail and fixes

SurgeMail 3.8k2 20-August-2007 (Webmail 3.1s-6)

- Security: IMAP buffer overrun (affecting logged valid email accounts only)

SurgeMail 3.8k 26-June-2007 (Webmail 3.1s-6)

- New: SPF additional option 'nohard' and g_spam_grey_nohard to allow SPF to be softened further.
- New: Made blogs lists sortable
- New: Added mouse click handling to webmails address autocompletion functionality
- Fix: Report generation failure if authentication db entries were larger than 1KB
- Fix: Webmail crashing bug if retrieving external accounts and mailbox was empty (recently added)
- Fix: g_archive message extraction (via tellmail archive_extract) was failing when an external filter had modified the message content
- Fix: Variety of other minor fixes

SurgeMail 3.8i3 10-May-2007 (Webmail 3.1s-4)

- Fix: Webmail version numbering tidyup. Webmail 3.1s-1 in surgemail build 3.8i2 was patched for security fix but version number not updated. To be absolutely certain you have the security fix make sure you are running the webmail executable version 3.1s-4 (either from surgemail build or standalone webmail build)

SurgeMail 3.8i2 9-May-2007 (Webmail 3.1s-1)

- Security: Webmail security fix (all versions of webmail should be upgraded)

SurgeMail 3.8i 26-March-2007 (Webmail 3.1s-1)

- New: Enhanced blogs security settings to limit blog comment spam (+ new setting [g_blogs_cleanup_links](#))
- New: Verification images (primarily used in blogs) are now distortion based (+ new setting: [g_verify_image_hard](#) - to make images even harder to crack - and read)
- New: Webmail addressbook autocompletion (off by default, enable with webmail.ini 'enable_autocomplete 1')
- New: Clicking "find config setting" search links highlights the relevant settings
- New: [g_proxy_webmail](#) - Redirect user.cgi logins to external host name
- New: Allow ||remoteip|| macro to be used in g_smtp_noauth
- New: Added support for 'user' token in "max_in" (eg.max_in="200k")
- New: [g_bind_byfromip](#) - Bind outgoing connections by sender ip
- New: Windows avast obeys g_vpipe_skip
- New: Extended "tellmail finduser" syntax to support wildcards etc
- Fix: (NASTY) If older than "n month" user.cgi mailbox rule processing crossed year boundary matched wrong date
- Fix: Webmail inbox corruption where some messages on list pane would get subject and addressing info from other messages
- Fix: Webmail null messages in inbox after login fixed
- Fix: Webmail fix WYSIWYG disabled for windows Vista (as MS [no longer supports](#) method used by webmail)
- Fix: Webmail sporadic bug that webmail would log out immediately on first login
- Fix: Webmail would eat the first characters of headers (of particular note the From header) if header did not have a space after the colon
- Fix: Webmail fix to make the display of mht messages more sensible
- Fix: Webmail smooth template only fix that could delete messages instead of move them under an obscure set of actions on the "move to folder dropdown".
- Fix: Webmail - attachments sent via webmail had a CR/LF appended at end
- Fix: IMAP import date handling so date is correct on migrated messages
- Fix: IMAP import bug that would fail to import folders if the destination server sent list responses with a trailing space
- Fix: IMAP import fix where sometimes processed folder count would get incorrectly displayed as zero
- Fix: IMAP import folder subscription handling if account already existed
- Fix: Domain admin log file search fix (on the first 9 days of the month, on unix systems, the domain admin msg.log delivery log search would not match any messages)
- Fix: Under certain conditions 'and' rules were being processed as 'or' rules
- Fix: Domain selection, if more than g_domain_list_max domains, on spam control, redirection, migration page
- Fix: Mailing list delete member handling if member has '_' character in name.

- Fix: Mailing list member sorting was broken about 6 months ago
- Fix: g_smtp_fix_nohead was broken in some cases
- Fix: default value handling with g_user_status_send
- Fix: Deletion of exception rules containing a quote character
- Fix: Removal of domain settings on mirrored host in some circumstances
- Fix: Sporadic crashes in domain keys handling
- Fix: Limits large file uploads under blogs
- Fix: FProt dot stuffing fix
- Fix: If g_send_speed was <500 bytes/s large messages would timeout due to buffering
- Fix: Minor memory leak in SSL connection handling
- Fix: Bug in signup per day limits
- Fix: sendmail_surge.ini handling fixed

SurgeMail 3.8f3 12-January-2007 (Webmail 3.1r-13)

- New: Blogs ping timeout added
- Change: Reduced stack size limits (reduction of max redirection loops)
- Fix: 4 fixes relating to uncommon crash conditions.

SurgeMail 3.8f2 8-November-2006 (Webmail 3.1r-13)

- Fix: Fixes related to g_include, and use of g_default_domain

SurgeMail 3.8f 30-October-2006 (Webmail 3.1r-13)

- New: [g_mirror_nwauth_always](#) - Always mirror nwauth even if using other authentication module
- New: Changed default install to set recommended settings on by default
- Fix: fix to dlist search command
- Fix: possible timeout problem due to invalid responses to ehlo requests
- Fix: Several fixes that could cause crash in unusual circumstances
- Fix: ndb performance related fix
- Fix: Log file searching smarts fix (recently broken)

SurgeMail 3.8d 13-October-2006 (Webmail 3.1r-13)

- New: [g_friends_confirm_debug](#) - Allow debugging of friends confirmation messages by logging friends actions to friend_confirm.log.
- New: mailing list "modify members" can now search based on user settings
- New: Add senders from ip address to relevant users log messages
- New: tellmail "-q" command line option added to run in quiet mode only outputting result of commands
- New: Copy and edit added to user filters and exceptions page
- New: [g_mirror_nwauth_always](#) - Always mirror nwauth even if using other authentication module
- New: [g_surbl_skip_ip](#) - Skips surbl lookups if the sender is from a listed ip
- New: Support for syslog server using [g_log_syslog](#), [g_log_syslog_host](#), [g_log_syslog_only](#)
- Fix: The way the date is stored on mailing list archives changed so that moving files with tar can preserve date information
- Fix: Domainkeys signing fix
- Fix: mailing list fix such that web_hide_email woks in show_headers mode
- Fix: Truncation of long lines on domain admin log file searching page
- Fix: Blogging of images fixed (recently broken)
- Fix: Friends template message header fix
- Fix: Domainkeys multiple txt record lookups (eg as used by yahoo.com)
- Fix: Viewing of html messages cpu loop fix
- Fix: Date displayed for SPF user.cgi log lines correct for front end / backend systems

SurgeMail 3.8b 28-August-2006 (Webmail 3.1r-12)

- New: [g_sms_forward](#) - This setting allows you to specify IP's which can send to an sms="TRUE" gateway without being authenticated. Normally only smtp authenticated senders can use these because we need to be sure the sender really is the person sending, not someone faking their address (in order to decrement the correct users quota). However, if you can already guarantee the sender, or you don't care then you can use

this setting. This allows your list server (dlist) to send to SMS numbers, for example.

- New: [surgewall_auth](#) - user="username" pass="password" - Makes SurgeMail use SMTP AUTH when delivering to the SurgeWalled server.
- New: [g_spam_share](#) - This lets surgemail share whitelisting information to improve scoring and avoid annoying spf bounce issues
- New: [g_friends_always](#) - Allows user's friends setting to be overridden
- New: Ability to view messages from user.cgi mailbox page & admin interface (requires [g_user_mail_view](#)) to see whether they are spam etc
- New: Additional settings to tweak SPF operation: [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_enforce_local](#)
- New: [g_blogs_domonly](#) - Make listed blogs page only show blogs in same domain as logged in user
- New: tellmail append_ini and append_lists command to append ini file sections
- New: "tellmail lookup recipient@domain" shows whether recipient is user / alias / mailing list etc.
- New: Ability to restore deleted domains (can restore domain + nwauth entries + users mail, not any originally deleted domain settings, aliases and mailing lists)
- Fix: g_orbs_check_all "true" caused g_orbs_late "true" to fail.
- Fix: was not correctly stamping when g_orbs_list was setup to deal with multiple responses. eg g_orbs_list name="combined.njabl.org" action="stamp" stamp="3=NJABL-DUL~4=NJABL-Spam~NJABL-other"
- Fix: g_orbs_check_all "true" caused g_orbs_late "true" to fail.
- Fix: Encoded email headers now get correctly decoded prior to filtering rules getting applied
- Fix: Send channel thread creation leak (freebsd only)
- Fix: Skip spf / grey bounce user setting related bug
- Fix: (g_)user_status_send not getting properly applied in certain conditions
- Fix: Blogs related UTF-8 charactr encoding fix
- Fix: recently added overzealous validity checking on redirect_cc fixed
- Fix: g_domainkeys_sign cr/lf bug
- Fix: Exceptions / filters page behaviour for surgewall domains

SurgeMail 3.8a 27-July-2006 (Webmail 3.1r-8)

- New: [g_forward_attach](#) - SurgeMail will send emails as attachments for specified domains when using late forwarding.
- New: SurgeMail supports CIDR format in ip based settings.
- New: [g_domainkeys_check](#) - Check incoming DomainKeys signatures (beta)
- New: [g_domainkeys_sign](#) - Sign outgoing messages (create a key first using web admin)
- New: [g_domainkeys_selector](#) - Policy name for your server (used creating dns entry for domainkeys)
- New: [g_domainkeys_only](#) - Domains to sign for outgoing email
- New: [g_user_friends_log_disable](#) - This disables the logging of lines to the users friends.log files (the users "log" page).
- New: [g_friends_old](#) - restores old friends behaviour
- New: [g_spf_byweb](#) - Description below in g_friends_byweb
- New: [g_friends_byweb](#) - These settings enable the use of a link to "allow" bounced and/or friends pending messages. The link returns a page with a verification image, the user enters the number shown and clicks "Add IP" or "Release message" depending on the context. In the case of spf allow, this link replaces the allow email address in the bounce message error string. In the case of friends, this link is added to the friends confirmation message (default) and the option to add it to the custom message appears on the custom message page.
- New: [g_late_forward](#) - described below in late_forward
- New: [late_forward](#) - These force late forwarding and hide the user option on the forwarding page that chooses whether to get it or not. If the user has already selected late forwarding setting either of these will not clear that, even if they load and save the forwarding page. Disabling these settings later will revert it back to the users previous personal setting as you'd expect.
- New: tellmail list_rcpt - Attempts to list all valid receipt addresses for the server
- New: tellmail dropfile_import user@domain c:\inbox\fred\mail - This looks for 'dropfile's one per folder, and creates matching folders/messages in surgemail (maildir format)
- New: tellmail bull_fwd [domain] - This searches the authent database, locates all the accounts which forward to a non-local address (and do not also deliver locally or deliver to a robot or responder) and emails them all the bulletins they have not yet received.
- New: Support for multiple webadmins added - "surgemail -admin_add"
- New: added mirroring of mfilter.dat and simple.rul files

- New: domainkey web admin interface page added
- New: [g_msg_log_extra](#) - The setting causes extra logging in the msg.log files, it logs user logins and imap fetch commands (to match the existing pop retr ones).
- New: You can now setup an auto response message and have it automatically turn off after a set period of time.
- New: tellmail can now use SSL for connections - use -ssl on the command line.
- New: forward message if smaller than xx (added to end user interface). If a message arrives which is larger than the specified size, it's not forwarded, it's delivered locally and a notification message is sent in it's place. This feature is especially useful to people who are forwarding email to pda devices with limited storage.
- New: "match exactly", "does not contain", "does not match exactly" in filters in end user interface. This feature only works if you use late forwarding (the interface makes this clear).
- Change: [g_forward_illegal](#) - Rules only apply to non local domains now.
- Change: [g_spam_allow](#) & [g_con_perip_except](#) changed to multi settings.
- Change: Instead of sending a confirmation (to a possibly faked address, which would then be considered spam and cause you to be blacklisted by spamcop or similar) it gave a 554 error. It did this if the message did not contain spf headers with "pass" in them. In other words it did this when spf was disabled or when spf failed. It has been changed so that it only does it when spf is enabled and fails.
- Change: On startup SurgeMail scans the mail queue, if this takes more than 10 seconds it skips the rest and makes an estimate.
- Fix: g_user_alias_file - strange problems when editing in webadmin when alias.dat does not exist
- Fix: redirect/redirect_cc - crashed when editing these settings in webadmin in some versions
- Fix: memory leak in cluster resync
- Fix: aspm performance
- Fix: dlist crash
- Fix: IMAP problem with messages with bad headers
- Fix: Some minor mirror problems.
- Fix: IMAP problem with changing folders caused problems for clients.
- Fix: IMAP codes when moving messages to inbox (improved efficiency)
- Fix: crash during ASPAM retrain.
- Fix: Mirror fix that caused some of the users settings not to be mirrored (bug was added in 3.7xx)
- Fix: rare crash in *nix in thread creation code.
- Fix: imapbody structure which most clients ignore except a few on osx
- Fix: [g_spam_allow_rbl](#) - wasn't working properly
- Fix: problem with quotas drifting upwards
- Fix: added more error checking to SurgeMail to detect admin mistakes like creating loops with redirect rules or adding domains that are already aliased to other domains etc.
- Fix: crash in domain keys code.
- Fix: rare admin bug that stopped the admin showing any values for settings.
- Fix: If machine A had a unique setting then changes on machine B would cause that setting to be removed from the config on A.
- Fix: DNS invalid domain - If a dns lookup failed due to a timeout, then the next lookup for the same domain would fail as if the domain didn't exist instead of with a timeout error.
- Fix: Pop proxy bug - When using gateway setting(s) and g_proxy_to_gateways the pop proxy was not being used for users in domains which do not exist in the config file (i.e. all of them)

SurgeMail 3.7b8 31-May-2006 (Webmail 3.1r-8)

- Update: Reverted SPF + Friends bounce behaviour (to enable feature now use g_responder_safer and g_friends_spf_fail_bounce)
- Fix: Mirroring of user cgi setting files (broken in most 3.7 builds)

SurgeMail 3.7b7 25-May-2006 (Webmail 3.1r-8)

- Fix: Several memory leaks
- Fix: Blog post editing and plaintext email posting fixes
- Fix: Footer & g_gateway_dat fix

SurgeMail 3.7b6 28-April-2006 (Webmail 3.1r-8)

- Fix: Netauth memory leak

- Fix: Quota handling uncommon crash
- Fix: Outlook IMAP folder redownload issue
- Fix: Friends related uncommon crash

SurgeMail 3.7b5 19-April-2006 (Webmail 3.1r-8)

- Fix: Friends store/pending view html email crash
- Fix: Webmail basic html message non display issue
- Fix: Mailing list archive viewer memory leak
- Fix: "all domain" reporting used default domain only
- Fix: Dlist hide_sender crash on linux
- Fix: Webmail HTML editor "press SPACEBAR or ENTER to activate and use this control" issue

SurgeMail 3.7b3 6-April-2006 (Webmail 3.1r-8)

- Fix: Aspam retraining performance bug
- Fix: Crash on freesbd on certain header addition operations

SurgeMail 3.7b 7-March-2006 (Webmail 3.1r-8)

- New: [g_block_skip](#) "address@domain" - exception for [g_block_files](#), for sender and rcpt.
- New: [g_header_strip](#) "header" - removes the header from the message.
- New: [g_imap_capa](#)
- New: [g_imap_capa_strip](#)
- New: [g_authent_last_login](#)
- New: [g_bounce_bind](#)
- New: [g_create_user_length](#)
- New: [g_iplimit](#)
- New: [g_iplimit_local](#)
- New: [g_iplimit_remote](#)
- New: [g_iplimit_islocal](#)
- New: [g_iplimit_whitelist](#)
- New: [g_user_send_rule](#)
- New: [g_language_default](#)
- New: [g_archive_files](#)
- New: [g_bounce_reject](#) "ip" - which you use to list the ip addressees of dumb back end servers that should not be allowed to send you bounce messages.
- New: [redirect_max](#)
- New: [language_default](#)
- New: [g_newui_advanced](#) "true/false" - can set the UI into advanced mode permanently, default false.
- New: [g_from_relay](#) & [g_from_relay_white](#) - if (relay_allow_ip matches) if (!authenticated) if (!islocal(from)) if (!whitelisted(from) bounce "sorry you need to be smtp authenticated to send from a non local domain"
- New: [g_user_utoke_expire](#)
- New: [g_forward_illegal](#) - allows you to stop users from setting up forward rules to certain domains
- New: Installer adds link to help
- New: [dlist_rotate_n](#) "n" - how many logs to keep for dlist (default 4)
- New: image based human verification for blogs
- New: popfetch now reads bcc's when using it with [g_spool](#)
- New: popfetch now tries to use the rcpt from received headers
- New: [create_image](#) "true/false" - this setting adds a verification image to the user signup process.
- Update: [g_spf_skip](#) is now a multi setting
- Update: removed some netwinsite URL's from surgemail/webmail as were causing problems with SPAM sites.
- Update: bounces now are never generated due to exception filters unless the setting [g_bounce_some](#) "true" is added
- Fix: timezone changes on Windows platforms should now work properly without restart. (daylight saving for example)
- Fix: when changing a group name, surgemail now updates all settings relating to that group correctly.
- Fix: popfetch in SSL mode.
- Fix: loading of mailing lists
- Fix: various crashes

- Fix: mirroring of files > 100 megabytes.
- Fix: problems with outlook and uid errors in imap
- Fix: display quotas of over 2 gigs
- Fix: problem with messages bypassing friends with an empty return-path <>
- Fix: renaming user from one domain to another domain

SurgeMail 3.6f7 1-December-2005 (Webmail 3.1q-2)

- Fix (3.6f7): Self account creation password problem
- Fix (3.6f7): Invalid message encoding confusing webmail fix
- Fix (3.6f7): Memory leak on quota report (if report was limited in some way)

SurgeMail 3.6f5 11-November-2005 (Webmail 3.1q-2)

- Fix (3.6f4): Unix intercept migration fix
- Fix (3.6f4): Webmail invalid UID fix
- Fix (3.6f4): Removal of several netwin urls in friends / bounce messages as this was getting servers blacklisted
- Fix (3.6f5): Bug with use of g_aspam_headers (unix only, introduced 3.6f4)

SurgeMail 3.6f3 4-October-2005 (Webmail 3.1q-2)

- New: Date range based reporting added to "tellmail report" per user / domain monthly reports.
- New: Integral support for blog listing with technorati.com
- New: Optional password protection for blogs
- New: Fancier surgemail bulletin features
- New: Account rename functionality
- Fix: Fixed non functional blogs_per_user_max
- Fix: Fixes to updated mailing list interface
- Fix: several other fixes
- Fix (3.6f2): Config mirroring and IMAP fix
- Fix (3.6f3): Webmail login issue (bug added post 3.6d)

SurgeMail 3.6d 9-September-2005 (Webmail 3.1p-8)

- New: New g_admin_access feature allows customisation of domain administrator features (per server / domain / per account)
- New: Webmail now handles replying to html messages in HTML without messing up html - in particular allows replies to HTML messages sent by MS Outlook (requires "allow_html_reply true" and "allow_style true" in webmail.ini)
- New: Per domain & per group log creation limits
- New: Posted blog images can be rotated on edit post page
- New: migration_translation.txt file support for handling weird account anames when using old_pophost / old_imaphost migration
- Fix: HTTP timeouts changed from 5s to 60 s
- Fix: New admin UI now has global use defaults (on accounts page)
- Fix: New admin UI was not saving surgehost.ini settings for default domain
- Fix: domain_defaults.txt works for domain creation under new web admin UI
- Fix: Quite a lot of other fixes :-)

SurgeMail 3.5b3 15-August-2005

- Fix: Blogs mirroring fix
- Fix: Blogs possible CPU loop when posting comments
- Fix: New admin UI fix - pressing save in domains page broke surgewall / user_sms_quota setting values
- Fix: New admin UI fix - vlist based settings automatic sorting conforms to old admin UI (affects g_access_group, g_download, g_header_out, g_include, g_retry_rule, g_sample_get, g_sample_show, g_send_tolimit not sorted)
- Fix: Webmail logins fail after upgrade (if first install prior to surgemail 1.5d and url_host is used)
- Fix: Back links removed in webmails use of surgemail web pages
- Fix: Perflog crash if system performance library did not return correct processor count (windows only)

- Fix: Crash if mail delivered to already full mailbox (solaris only, recently introduced)
- Fix: tellmail quota_rebuild "invalid account" fix (bug introduced surgemail 3.2a)
- Fix (3.5b2): Cosmetic new admin UI fix (removal of warnings introduced by fix above)
- Fix (3.5b3): Incorrect addition of g_authent_domain "true" if no setting present (not using g_authent_domain) and saved from web admin ui
- Fix (3.5b3): New admin UI bug, save on domain page in standard mode reset mailbox_path to default value
- Fix (3.5b3): HTTP timeouts changed from 5s to 60 s
- Fix (3.5b3): Couple of new admin ui and blogs cosmetic changes
- Fix (3.5b3): Bug that could delay messages by a few minutes if send_isslow feature is used - default in this version
- Fix (3.5b3): Aspm filtering fix

SurgeMail 3.5a 26-July-2005 (was 3.2g 19-July-2005)

- New: Additional wave template set for blogs
- New: Additional dlist settings: post_members_tomod, mod_hide, strip_headers
- Fix: Several minor new web admin UI fixes
- Fix: Sending fix if remote system always hung while sending headers
- Fix: NFS efficiency fixes
- Fix: Memory allocation efficiency fixes
- Fix: Minor blog fixes
- Fix: Mirroring fixes (now mirroring pending.dat and custom.dat)
- Fix: Several minor surgeplus fixes
- Fix (3.5a): Delete domain button added to new ui + several new UI fixes.

SurgeMail 3.2e 1-July-2005

- New: New simplified and more informative UI for SurgeMail Admin web interface
- New: Surgemail Blogs allowing users to create and manage blogs
- New: Many minor features added, too many to list individually
- New: [suspend](#) "true/false" - allows suspending the domain, allows incoming mail, but denies the ability of users to check their mail.
- New: [g_fix_crCrLf](#) "true/false" - setting to deal with some legacy applications sending incorrect cr/lf's at the end of headers
- New: [g_auth_norelay](#) "true/false" - if true does not let user relay because they authenticated
- New: sendmail.ini file in (g_home directory) controls some aspects of sendmail
- New: [g_smtp_portauth](#) - Forces smtp authentication on this port eg g_smtp_portauth "587"
- New: [tellmail_add_rules](#) <file>
- New: [g_send_nolimit](#) <domain> - Overrides g_send_max_perdom for one or more domains
- New: Allowed changes to SSL certificates to take effect without a restart
- New: [g_tarpit_retry](#) - so that tarpit limit results in a retry rather than a 5 second delay
- New: [g_msg_max_hops](#) "number" - sets maximum number of received lines before rejecting (default 30)
- New: email notification message - The first 1024 chars of the body are now available with ||body||
- New: [g_quota](#) group=string quota=string
- New: [tellmail_add_member](#) <list> <email> - adds member to mailing list
- New: [g_smtp_noauth](#) "ip" - all non matching ip's must authenticate
- New: [alias_max](#) - limits the amount of aliases allowed in a domain
- New: [g_allow_passzip_to](#) "..." - allows avast to let through unmonitorable zip files to address
- New: [g_allow_passzip_from](#) "..." - allows avast to let through unmonitorable zip files from address
- New: [g_orbs_report](#) "true/false" - This allows you to get an email if your server gets 'orbs' listed
- New: [g_include](#) / [g_download](#) - Used together you can fetch a file from a web server, and get updates hourly,
and include it for global settings in your surgemail.ini
- New: [g_body_filter](#) "true/false" - end users can now choose to filter on the body of the message
- Fixed: [g_vpipe_skip](#) "ip" - now skips [g_virus_fprot](#)
- Fixed: sendmail crashing when using sendmail.ini
- Fixed: Deletion of domains (changed to two actions, delete then purge)
- Fixed: Many minor bugs
- Fixed: Various bugs that could cause crashes
- Fixed: Many cosmetic issues

- Fixed: Deleting in outlook using imap with multiple threads
- Fixed: Updated IMAP IDLE command to timeout after 32 mins
- Fixed: Random bug where a 5 second delay could be caused between WebMail and SurgeMail
- Fixed: Bounce to rules were being lost in some situations
- Fixed: Many cross site scripting fixes
- Fixed: DNS UDP runt packets being sent
- Fixed: Added code to check when tarpitting channels that there are always enough channels left for connections - if not old ones are dropped.
- Fixed: g_bad_from doesn't block if it gets a 400 error now as was blocking emails due to greylisting
- Security: [g_web_max](#) "100" - maximum connections
- Security: [g_web_max_perip](#) "10" - maximum per ip
- Security: [g_deny](#) "ip" is now used by webconnections also

SurgeMail 3.0c2 22-March-2005

- Security: (2.2c2 : 22 March 2005) Webmail security issue and user.cgi security issue fixed.
- Add: Updated version of webmail to webmail 2.1n (build23)
- Note: This release is really just what would normally have been 2.2g5 (does not contain features in 2.2h builds)

SurgeMail 3.0a 17-March-2005

- Fixed: 5 second delay affecting webmail running under surgemail (affects several unix versions only)
- Note: This release is really just what would normally have been 2.2g4 but with a major version number update (does not contain features in 2.2h build)

SurgeMail 2.2g3 5-March-2005

- Fixed: Remove the option of deleting domains from the admin as it some circumstances it could remove the wrong domain (two admins using the admin at the same time, or an admin with two browser windows open to the admin)

SurgeMail 2.2g2 15-February-2005

- New: [g_surbl](#) - Takes advantage of databases for known spam URL's (www.surbl.com)
- New: [g_rcpt_nodup](#) "true/false" - Stops duplicates of the same message being delivered multiple times to one address (only in same session) - works well when using fallbacks etc.
- New: Extended archiving flexibility - see archive section in webadmin.archive
- New: Can edit members in a mailing list - disabled true/false, onholiday true/false, digest true/false
- New: [g_orbs_list](#) - This setting can now have different stamps based on the ip returned by the RBL.
- Fixed: problem where a forward to multiple users with one user over quota gave back a meaningless error.
- Fixed: blank gateway rules caused problems.
- Fixed: cleaned up various logging messages.
- Fixed: size() in mfilter not working with values over 14999.
- Fixed: Many other unspecified fixes
- Fixed: (2.2g2 : 15 February 2005) Surbl rule formatting bug
- Note this build contains the same version of webmail as the surgemail 2.2c10 release.

SurgeMail 2.2c10 17-January-2005

- Security: (2.2c9 : 23 December 2004) Webmail security bug fixed.
- New: [g_smtp_welcome_delay](#) "seconds" - delays smtp welcome, any connection that doesn't wait for prompt before sending data is dropped.
- New: [g_orbs_cache_life](#) "seconds" - Time to keep RBL hits cached
- New: [g_orbs_force](#) "true/false" - forces rbl lookups even if user is in an allowed relay ip ([g_relay_allow_ip](#))
- New: tellmail [surgehost_update](#) - makes surgemail update surgehost.ini for each domain.
- New: [user_list_quota](#) - This setting configures the number of mailing lists a user of this domain can create
- New: [g_user_list_quota](#) - This setting configures the number of mailing lists a user can create on this server
- New: [g_admin_guesses](#) "number" - number of guesses allowed for admin login before being shut out.
- New: [g_user_send_max](#) "number" - Allows the specification of a maximum number of emails per day.
- New: [user_send_max](#) "number" - Allows the specification of a maximum number of emails per day.

- New: [g_user_filter_early](#) "true/false" - this setting causes it to run the users exceptions/filters before writing the message to disk during the delivery process.
- New: Multilanguage support for user.cgi pages
- Fixed: autoresponders failing when used with [g_orbs_late](#) "true"
- Fixed: memory leak
- Fixed: quota handling problem on surgemail crashes
- Fixed: dlist - quoted text before the address was not handled eg "text here" <address@domain>
- Fixed: (2.2c7 : 8 December 2004) Friends message pending list did not display email address correctly.
- Fixed: (2.2c10 : 17 January 2005) 3 fixes - crash when adding non local dlist members, minor memory leak, and debug filehandling fix.

SurgeMail 2.2a6 1-November-2004

- New: DNS lookups now use UDP.
- New: Per user usage logging for gatewayed / surgewalled domains + more report generation features
- New: [g_mailstatus_message](#) state="string" message="string"
- New: [g_send_helo_from](#) "txt" - Uses from envelope to generate helo.
- New: [g_received_skip](#) "true/false" - Skips writing the received header for trusted local user.
- New: No rbl lookups are done if the user matches g_relay_allow_ip
- New: [list_max_users](#) "number" -sets maximum amount of users that can be added to the lists in this domain
- New: tellmail change_pass <user@domain> <new password> - changes users password.
- New: CRAM-MD5 support added - Documentation [here](#).
- New: [header_add](#) <header> - adds headers to messages, domain setting.
- Fixed: installer for FreeBSD systems.
- Fixed: The imap command asd fetch 1:* BODY.PEEK[HEADER.FIELDS.NOT (RECEIVED)] imap returning incorrect data
- Fixed: IMAP IDLE command.
- Fixed: delay in reading in dns.dat cache.
- Fixed: File leak on avast update.
- Fixed: perflog stability
- Fixed: handling of MIME messages and footers in dlist.
- Fixed: join template and naked line feeds in dlist.
- Fixed: blocked all ips from admin interface if 6 incorrect attempts were made, now done per ip.
- Fixed: g_spam_allow if blank allowed everyone.
- Fixed: Variety of other minor fixes

SurgeMail 2.1c7 9-September-2004

- New: Improved DNS lookup code
- New: [g_orbs_late](#) "true" - allows late lookups so authenticated users can be bypassed for rbl checks
- New: [g_spf_skip_to](#) "user@address" skips spf checks for specified rcpt, also skips rbl checks if using g_orbs_late "true"
- New: [g_quota_skip](#) "ip" - skips quota checks for users sending from specified ip
- New: [g_filter_n](#) "number" - Number of filters to run simultaneously
- New: Added more information to the status pages (mainly bandwidth related) and status page readability improvements
- New: Improved surgeplus integration
- New: Install dbabble from within surgemail web admin interface
- New: Latest version of webmail now included with surgemail releases (for this release v3.1k build 8) for webmail release notes see <http://netwinsite.com/webmail/updates.htm> - part of webmail admin manual
- New: Improved web image caching for surgemail and webmail
- New: Build number now included as part of filename on patched builds (instead of separate patch number) to avoid version confusion
- Fixed (2.1c7): DNS bug rejecting some emails "DNS channels all busy for 5 minutes"
- Fixed (2.1c7): Per domain/user logs deleted based on g_record_days
- Fixed (2.1c6): Further perflog crashing bug fixed
- Fixed (2.1c6): DNS display on standard status page
- Fixed (2.1c6): Per user / domain reporting did not work if g_record_path was used
- Fixed (2.1c5): Continuous peflog problems fixed by making sampling code more robust

- Fixed (2.1c5): Webmail fix - Smooth template set reply etc would not send
- Fixed (2.1c5): DNS lookups on surgewall domains
- Fixed (2.1c5): Several other minor fixes
- Fixed: The uninstaller so it removes all files and rollback mechanism works correctly if file busy error encountered during upgrade
- Fixed: Some confusing SMTP error messages like this one "failed (Success)"
- Fixed: Bug in the sendmail stub that caused it to crash with the -mailfrom argument
- Fixed: Several stability bugs
- Fixed: Peflog restarting surgemail if missing more than 5 minutes of data
- Fixed: Bug when sending with g_spool, sometimes mail got sent multiple times
- Fixed: SMS notification quota handling
- Fixed: Several changes + fixes to SPF handling
- Fixed: Mailing list handling fixes

SurgeMail 2.1a 6-August-2004

- Fix: Perflog graphing bandwidth_in and bandwidth_out graphing could get broken sometimes
- Fix: Several stability issues and a few other minor issues
- 2 new webmail template sets (Surge and Smooth), both with multilanguage support (English, French, German, Spanish, Portuguese)
- SPF improvements and new settings (g_spf_default, g_spf_default_noblock, g_spf_skip, g_spf_rev_skip)
- Usage reporting for usage / accounting purposes (per user and per domain mail received / send / pop usage etc)
- Latest version of webmail (version 3.1i build 13)
- Some surgemail admin web interface improvements
- Viruses stopped by all mechanism now logged on advanced status page
- (non functional - Windows only) On fault stack is now dumped by surgemail to startstop.log instead of drwatson logs for catching crash information as drwatson logs not reliable on Windows 2003

SurgeMail 2.0g2 10-July-2004

- Fix: (2.0g2): Using g_access_groups with access_smtp not matching IP prevented receiving instead of sending of messages (broken 2.0c)
- Fix: Sporadic crash due to perflog graphing
- Fix: cc's in mfilter when entered via the webadmin
- Fix: UID issues in imap especially when used with outlook
- Fix: fixed problem with reloading mfilter, could cause crashes.
- Fix: Using our own dns code on freebsd now as theirs was buggy - reverse dns lookups will work etc.
- +: Added domain quotas - setting quota_domain "500mb"
- +: Improved logging for RBL hits, logs which RBL was hit
- +: Status page now shows how many hits for each RBL you are using

SurgeMail 2.0e 10-June-2004

- Fix: search feature in surgemail now a lot faster
- Fix: spf templates now use "strict" not "mean"
- Fix: g_mfilter_skip_ip problem fixed , caused skipping if no value was set
- Fix: tellmail logout - wouldn't logout users in certain states.
- Fix: dns problems related to SPF lookups.
- Fix: friends/exceptions/spam problems fixed (added in 2.0b)
- Fix: g_tellmail_ip "ip" now works.
- g_dsn_enable "true/false" - default false, must set to true to turn on dsn support.

SurgeMail 2.0c 3-June-2004

- Fix: Now rejects messages on receipt of the SIZE declaration
- Fix: Problem when daily logs get too large (> 2 gigs)
- Fix: IMAP dying on certain messages
- Fix: Fixed some tellmail commands that were not working properly (add_user, delete_user etc)
- Fix: Problems with user admin buttons not working.
- Fix: Friends release didn't move messages to inbox.

- Fix: Improved stability
- Fix: stopped displaying full paths for various file errors (thanks to www.exploitlabs.com)
- Fix: Improved speed for systems with large amount of domains
- +: added support for #include's in surgemail.ini - won't be able to use webadmin for saving changes however.
- +: IMAP4 UIDPLUS extension added (rfc2359)
- +: g_mfilter_skip_ip <ip> skips mfilter for messages coming from selected ip's
- +: added support for x-receiver headers for g_spool_path
- +: Added DSN's
- +: tellmail suspend - stops surgemail from delivery mail, just queues it.
- +: tellmail resume - tells surgemail to start delivering mail again after a suspend command has been issued.
- +: Surgemail performance counter (and windows perfmon system counter) history graphing
- +: Webmail version 3.1g (includes security fix)

SurgeMail 2.0a2 21-May-2004

- Fix(2.0a2): Friends messages contained bad header (broken 2.0a)
- Fix:(2.0a2): Remote tellmail command using "-host" (broken 1.9b)
- Fix: dlist problems with subscribes/unsubscribes not working
- Fix: several security issues (not showing full paths on errors etc) (thanks to www.exploitlabs.com)
- Fix: patched ssl version to cover security issues in open ssl
- Fix: smitecrc was broken in 1.9 builds.
- Fix: dlist problems with digests not going out and missing messages
- +: g_spf_domain "domain" - specify the domain to use in spf and srs rewrite responses.
- +: g_retry_dns "hours" - retries messages that get a dns error (dns responds but with nothing valid).
- Fix: 5xx errors on connect now cause bounce instead of retries.
- Fix: setting bounce in mfilter rules now causes a bounce instead of scoring it high.

SurgeMail 1.9b2 19-May-2004.

- Fix (1.9b2): smitespam binary only updated to version built 19 May (installer and surgemail binary will still say version 1.9b. Alternatively just run 1.9b build (12 May) and update smitecrc from web admin)
- Fix: g_spam_allow now stops lookups for g_badfrom_check
- Fix: DNS problems, caused lookups to fail in some cases.
- Fix: smite_skip_ip was broken
- Fix: Improved stability
- Fix: account status's now work for smtp
- Fix: SurgeMail could crash while searching for a config setting.
- Fix: domain aliases with user aliases were broken.
- Fix: SurgeMail wasn't decreasing user license numbers in some circumstances when users were deleted.
- Fix: Aspm was being classed as a filter not a spam user access setting, now corrected to use the 'spam' access
- Fix: Fixed the infamous "399" tcp errors when unable to connect to other servers, now gives proper reasons.
- Fix: Can now set moderator password for lists in the webadmin.
- +: g_smtp_max_nolimit <ip> - Ip addresses that don't have max smtp limit applied.
- +: list_max <number> - maximum number of lists that can be created for that domain.
- +: list_disable <true/false> - disables list creation for that domain.
- +: added access_list to dlist, can control who sees particular lists with the "list" command.
- +: logging changes, now logs the setting that gave a user permission to relay and if they used smtp auth logs the account they authenticated with.
- +: tellmail add_domain <domain> - adds domain to surgemail
- +: tellmail add_user <user@domain> <password> - adds user to user database
- +: tellmail add_user_alias <user> <alias> - adds an alias for the specified user
- +: tellmail delete_user_alias <user> <alias> - deletes an alias for the specified user.
- +: g_orbs_list will now use the stamp as the rejection message in deny mode.

SurgeMail 1.8g3 25-March-2004

- Spool path allowing SurgeMail to send dropped files as email (g_spool_path)
- New settings (g_delete_user_mode, g_user_alias_file)
- SurgePlus calendar and filesharing (fully functional but still beta)

- Fix: Memory corruption bug causing sporadic random crashes on SSL connections
- Fix: Vpipe restart bug on timeout
- Fix: Further fix to new na_exceptions web page
- Fix: Minor memory leak
- Fix: Proxy mode fix
- Fix (1.8g2): WebMail autologin fix for non default vdomains using custom quick login method
- Fix (1.8g2): Alias creation bug
- Fix (1.8g3): Hopefully final fix to WebMail autologin. (Rolled back logic to that of previous release builds)
- Fix (1.8g3): Reports page broken on UNIX systems
- Fix (1.8g3): Bug in SurgePlus that could sporadically crash SurgeMail on display of SurgePlus web page

For updates prior to 1.8g3 see the [older surgemail change history](#) page.

SurgeWeb Change history

If you find a bug please report it to surgemail-support@netwinsite.com.

- **New:** Allow pinning of important messagers to the top of the message list. Can be based on all "Starred items", or messages with one or more Universal Labels (eg. "Important,Today"). Enable / disable using "sort by" menu and edit pinning labels on labels management page.
- New: Automatic interface reload if preferences have been changed that require a page refresh to get actioned (notably language & screen layout)
- New: Clicking stars to flag / unflag messages applies to the whole selection if multiple items are selected.
- Change: Removed the automatic screen refresh if you had actioned >50 items of the 100 items in the message list, as it was mostly "annoying" rather than useful
- Fix: Surgeweb frontend/backend user.cgi autologin now works with just g_surgeweb_backend_server specified (previously only worked with per domain setting surgeweb_backend_server)

Sat 8 September 2012: Specials build 6.1e-25

- Fix: Several minor tweaks

Wed 5 September 2012: Specials build 6.1e-23

- **New:** (BETA) Caldav integrated "drag and drop" calendaring from surgeweb ajax interface (see [key features](#)). The calendaring interface is beta and there are still features I am working on to make it easier to use (eg datepicker when editing etc). If caldav is [installed](#), surgeweb will display caldav calendar for all users. There is a setting on the surgeweb customisation page to set the global default to hidden. If globally disabled you can enable per account using "newcal_hide false" in _user.dat files. As it stands all critical bugs I know about have been resolved, it is very usable, and it is decently cross browser compatible (Very nice in latest IE9, Chrome, Safari, FF; ok in IE8; usable but displayed with old browser warning in IE7)
- Fix: Compose button (and rest of toolbar) was behaving oddly under IE7 only (recently introduced)

Wed 29 August 2012: Specials build 6.1e-15

- New: Surgemail (and surgeweb) standardised for quota handling in units of 1000 rather than 1024 ie 1mb=1000000 (means quota in admin interface can be set as 100000000 instead of 100mb and still match quota in surgeweb and mailbox full email)
- New: Allow specification of charset for old webmail contacts and distribution list imports, specified as surgeweb config*.dat setting of eg: "contact_import_charset ISO-8859-1" (should be set to match the display character set the old webmail interface was being used in for cleanest imports)
- Fix: Hide the "options - preferences - filtering log" link if spam and friends are disabled (as the behaviour used to be before recent spam related interface improvements)
- Fix: Support single quotes in signature name string
- Fix: "home" was being shown as "home page" in some of the contacts fields (introduced 6.1d-33)
- Fix: "There are no messages in this folder" sometimes incorrectly displayed (Pressing refresh or new mesasge arrival without having pressed a folder in this session - recently broken)

Tue 21 August 2012: Specials build 6.1e-9

- Fix: Two IE only javasript errors fixed

Thu 2 August 2012: Specials build 6.1d-41

- Fix: Long CC header crashing bug introduced in 6.1d-40

Tue 31 July 2012: Specials build 6.1d-40

- New: Delivery log and Spam control links added to the Folders pane. (can be removed via config*.dat setting of "no_spam_panel true")
- Fix: Cc headers (and Bcc headers of messages in sent folder) were not correctly decoding utf8 characters resulting in "?=iso..." in display strings.
- Fix: Split backend / frontend servers will now pickup the surgeweb interface language selection and apply it to the user.cgi page too
- Fix: Can remove context sensitive help suffixes using surgeweb config_*.dat setting of "help_context false"

Tue 20 July 2012: Specials build 6.1d-33

- Fix: Equals key stroke to move to done folder was not working in FF

- Fix: Lots more strings are now language translatable
- Fix: Additional checks on pref_sizer1 which was sporadically receiving negative values
- Fix: Issue where clicking messages was no longer opening them after having sent a message (only affected unusual screen layout combination)

Tue 3 July 2012: Specials build 6.1d-24

- Fix: Surgeweb was no longer working in IE7 after recent code changes. (other versions of IE & other browsers unaffected)

Mon 2 July 2012: Specials build 6.1d-23

- Fix: Display of simplified spam handling in the basic interface as well.
- Fix: If surgevault was disabled the lock was no longer getting hidden on popup windows.
- Fix: Increased the surgeplus web upload filesize limit from 100MB to 800MB per file.

Sat 30 June 2012: Specials build 6.1d-21

- New: Moving messages to Done or Trash folder automatically marks them as read if needed
- Fix: Fixes to the save handling of the "Note" messages
- Fix: Doing forward of multiple messages was only marking one as forwarded (same with the send-done on a forward was only moving the first message)
- Fix: Switching of identities with Ctl-Alt keys pressed was no longer preventing the mark-read actions.

Wed 22 June 2012: Release build 6.1c-1 (Existing beta distribution upgraded to release)

Wed 6 June 2012: Beta build 6.1c-1

- New: Standardisation / simplification in the presentation of the user.cgi / surgeweb spam configuration interface. Some changes to the recommended spam handling settings.
- Fix: Several stability fixes
- Fix: Google chrome would sometimes "stop doing its resize updates"

Thu 10 May 2012: Specials build 6.1a-1

- Fix: Faster display of the body of the contacts list when switching to contacts via app menu
- Fix: Folder switching with screen layout set to "Web" rather than "Application" mode was resulting in odd behaviour (result of recent new features)

Tue 1 May 2012: Specials build 6.0b-61

- Fix: Reply button on popup message windows broken with recently introduced features

Tue 1 May 2012: Specials build 6.0b-60

- Fix: Minor tweaks to the Notes handling
- Fix: Two minor fixes wrt caldav calendaring. (configuration via surgeweb for some configurations, behaviour if unix php timezone is not set)
- Fix: Pressing delete key during contacts editing was not working correctly since recent contact multiple selection changes

Thu 26 April 2012: Specials build 6.0b-59

- New: Surgeweb now adds the surgeweb reply coloring to the sent message when sending in html mode. Makes it look a whole lot more readable in microsoft clients in particular.
- Fix: Surgevault related fixes (surgevault lock displayed with surgevault set to hidden in surgeweb admin interface, and switching to surgevault with when deleting addresses - both introduced in recently)
- Fix: Replying to messages was not correctly adding images in signatures to the message

Tue 24 April 2012: Specials build 6.0b-56

- New: (non surgeweb) User.cgi domain admin logins now support multiple concurrent sessions
- Fix: Surgeweb CalDAV configuration interface now works with ip address specified in g_webmail_port or using surgeweb's https_required setting.

Tue 24 April 2012: Specials build 6.0b-55

- Fix: Improved the addition of "additional accounts" which had never been logged into directly using surgeweb. Notably was resulting in

folder could not be loaded popup boxes in recent builds and the warning "tcmd_load_msg_info no fld loaded 1" in slightly older builds.

Mon 23 April 2012: Specials build 6.0b-54

- **New:** Full multiple selection handling for manipulating contacts on contacts page (selection handling just like email mailbox message list). Mailbox code was generalised and reused so watch for anything I may have broken on the mailbox message list selection handling.
- **New:** Ability to save "Notes" in your email boxes - Useful for variety of things including: free format To Do lists, quick reference notes, inbox annotation, and general use instead of a Post-it note.
- New: Ability to modify messages in your Templates folder and save directly back to templates folder
- Fix: Much faster refreshing of the whole contact list (previously only hit at login time or deletion of contacts, now also hit when actioning multiple items)
- Fix: Can now delete contacts that have the "+" character as part of the email address (notably affected were auto added Facebook reply by email addresses)
- Fix: Reopening drafts (or notes) with attachments / creating message with attachments on compose (eg forward attach) was not correctly enabling the desktop drag and drop attachment integration
- Fix: Message list caching works fully across multiple accounts now too
- Fix: Mouse based autocomplete selection picking recently broken (IE8 only, IE9 & other browsers not affected)

Thu 18 April 2012: Release build 6.0a5-5 (main production release download links updated to this build)

- Fix: Windows distribution updated so avast should install now on the few machines where it was failing to install. (for already installed versions of surgemail you may need to delete or rename the surgemail/avast directory first).

Thu 5 April 2012: Specials build 6.0b-49

- Fix: Folder message list caching was broken in IE8 leading to odd folder switching behaviour.

Wed 4 April 2012: Specials build 6.0b-48

- Fix: Several minor fixes & tweaks to caldav calendaring (if caldav already installed you will need to press "update current install" using admin interface)

Wed 4 April 2012: Release build 6.0a5-5 (patched)

- Fix: OSX_intel build should now run on older Intel Core Duo macs again (in fact the osx intel build is back to being 32bit as there is little advantage for it to be 64 bit although there is a "FAT" combined 32 / 64 bit build next to the normal download if anyone does want to try it)
- Fix: Windows build is now "signed by NetWin"
- Fix: (non surgeweb) g_inbox_max related bug creating incorrect totals & empty messages in spam / friends folders
- Fix: (non surgeweb) fix for smtp sending crashes

Sat 31 March 2012: Specials build 6.0b-45

- Fix: Message list caching was not updating properly for newly added messages to folders with more than 100 messages (eg opening sent folder after sending a message)

Fri 30 March 2012: Specials build 6.0b-44

- **New:** (non surgeweb) Surgemail serverwide SSL SNI support providing the ability to host multiple domains with standard ssl certificates on the same ip address.
- **New:** Folder message list caching with list update as soon as a background update returns. Means switching between folders particularly with slow server or high latency network connection is more responsive. Rather nice side effect of this is that you effectively have a per folder cursor that gets remembered as you switch back and forth between folders. Clicking currently displayed folder in folder list still does normal full update. Enabled by default, users can disable in advanced preferences or set global default using config_*.dat of "pref_nocache2 true".
- Fix: Image customisation using surgeweb admin interface for additional vdomains on single url signon with customisation has never worked. It was incorrectly looking for images in surgemail/surgeweb/custom rather than surgemail/surgeweb/custom/domain.com.
- Fix: Surgemail installer (Windows only) now "signed by NetWin" rather than being "unsigned".

Wed 21 March 2012: Specials build 6.0b-39

- New: CalDAV calendars now in BETA. Calendar sharing / creation / deletion accessible via surgeweb. Can be shared with "write access" / "read access" / "free busy" ([further info on installation](#))
- New: Setting to disable the Flash uploader and always revert back to plain form based uploads
- Fix: Fixes to the contacts getting shown as coloured when adding addresses in the To / Cc / Bcc field

Fri 17 March 2012: Specials build 6.0b-35

- Fix: Address autocompletion was behaving oddly for html interface or with fancy address picker disabled in ajax interface (recently broken)

Fri 17 March 2012: Specials build 6.0b-34

- Fix: Nasty bug where surgeweb was failing to send and quietly eating most messages. Bug introduced last night in version 6.0b-32

Thu 16 March 2012: Specials build 6.0b-32

- Fix: Improved surgeweb compatibility talking direct to non surgemail backend smtp servers
- Fix: Prevent addresses that look like facebook etc dynamic for getting autoadded to the contacts list

Wed 7 March 2012: Specials build 6.0b-26

- New: Beta of surgemail hosted standalone CalDAV calendaring for mobile and desktop clients (using php based SabreDAV). Should be essentially functional (primarily tested using iOS clients), but do not deploy in production yet as data storage arrangement may be changed yet. Contact netwin if you are interested in beta testing this. Further features will be added here in the near future.
- Fix: Bug where single part messages were not getting correctly decoded for mime type. Notably visible as character set handling issue where single part plain text messages with native table based character sets were not correctly displayed as utf8. (introduced 5.3i-43)
- Fix: Crashing fix surrounding processing of really long urls in some emails
- Fix: Made switching between 32 bit and 64 bit binaries tidier in terms of surgeweb errors (notably affects 6.0a2-2 OSX intel build upgrading from older osx builds. But other 64 bit builds will regenerate users surgeweb indexes too with this fix)

Thu 1 March 2012: Release build 6.0a4-4 (patched)

- Fix: Further fixes in terms of upgrade of 64 bit builds. Fix only affects linux64, solaris8 & osx_intel builds and if you are already on 6.0a3-3 without problems there is no need to upgrade to 6.0a4-4.

Fri 24 February 2012: Release build 6.0a3-3 (patched)

- Fix: Spam filtering handling crashing fixes (already resolved on main branch)
- Fix: Made switching between 32 bit and 64 bit binaries tidier (further info noted above)
- Fix: Added "all_hints_disable true" setting (already part of the main branch)

Wed 22 February 2012: Specials build 6.0b-20

- New: Detected Adobe Flash version displayed as part of the info page under the green info icon.
- Fix: Crash in addressbook export.

Tue 14 February 2012: Specials build 6.0b-18

- **New:** Multipart related image support in the signatures to support "business card" type images as part of the signature.
- New: Emoticon icon support as multipart related inline images when composing a message
- **New:** Ability to add multipart related inline images when composing messages. Includes serverside rescale and crop facility as part of the upload dialog.
- **New:** Signature improvements - now supports multiple signatures, of unlimited size, with the ability to set the default for each additional account.
- Fix: Improved preferences saving, previously if a background request started earlier than and finished after a preferences save requests, the changed preferences could annoyingly be lost. (particularly noticibaly if making preferences changes straight after login while contact or messages were still getting downloaded).

Tue 14 February 2012: Release build 6.0a2-2 (Existing beta distribution upgraded to release)

Tue 24 January 2012: Specials build 6.0b-7

- **New:** Multiple account support within surgeweb. Many of you have been using this for a long time but this resolves most of the little gotches and oddities this feature still had. This allows you to have multiple local imap accounts within one surgeweb login (keeping them separate, but monitoring all for new mail, and able to reply from any account). Also allows you to specify the reply to header. Note: Intended and partially complete "single inbox" multiple identity support has been disabled for now until implementation is complete. Multiple account support can be completely disabled on the surgeweb customisation page if needed.
- **New:** Access to the Legal Archives though the use on an "Extras" tab in the preferences. Also provides access to other user.cgi functionality which was not yet available directly through surgeweb: notify, mailing lists, mail import, address aliases (only if those features are user_access enabled). Can also be disabled at surgeweb level with config_*.dat setting of "extras_disable all", extras tab features can be

individually disabled using this same setting.

- **New:** User selectable automatic logout if surgeweb session is idle for too long (can be forced on using config_*.dat setting "autologout_enforced true" and configured using say "autologout_duration 30", "pref_autologout_type on"
- New: Hints system recently introduced can be completely disabled with the config_*.dat setting of "all_hints_disable true".
- Fix: Some image attachments were not getting seen as image attachments

Mon 17 January 2012: Beta build 6.0a2-2

- Fix: Surgevault related crash resolved

Tue 10 January 2012: Beta build 6.0a-1

- New: Message save to file naming improvements. Zip file name is based on folder and export date, contained message file named based on message date, message subject and message sender.
- New: Surgeweb hints and suggestions for the productive use of surgeweb. A small set of hints now get occasionally displayed to surgeweb users. The default list of system hints can be customised with up to 32 of your own hint messages, and individual system hints can be disabled if necessary.
- Fix: BCC field was getting truncated to 1KB (approx 35 recipients) when reading in message - eg viewing of a sent message, or re-editing draft, etc

Sat 7 January 2012: Specials build 5.3i-64

- Fix: Surgeweb was sending messages with empty Message-ID headers, with primary result that a few receiving mail servers/clients were "deduplicating" and deleting the messages in question (broken 5.3i-35)

Thu 8 December 2011: Specials build 5.3i-59

- New: Surgeweb new message refreshes tweaked to also hide messages if they disappear. (as a result of getting processed with another mail client / phone etc). This will introduce some extra diskio, old behaviour can be restored using surgeweb config*.dat setting of "global_nohide true"

Tue 29 November 2011: Specials build 5.3i-50

- New: Ability to mark return path as valid for from address so additional warning header does not get displayed for future messages for this address combination.
- Fix: Drag and drop attachment addition works again for latest versions of firefox.

Fri 4 November 2011: Specials build 5.3i-40

- New: Extra header display tamed slightly to count same base level domains as valid matches by default. Behaviour can be configured with config_*.dat setting "mismatch_check" of disabled, domain or exact.

Thu 3 November 2011: Specials build 5.3i-39

- New: Extra headers displayed if messages seem dodgy (reply-to, return-path, reverse lookup) displayed under various combinations of header address mismatch, or message being identified as spam.
- New: Copy email address text via right click menu on any address recipient or right click menu in message list.

Tue 10 October 2011: Specials build 5.3i-36

- Change: Old webmail no longer upgraded or installed by default - is still part of the distribution (running installer with "-dowebmail" still installs old webmail)

Tue 23 August 2011: Specials build 5.3i-31

- Fix: Contact details would not allow you to specify Phonetic names at all anymore.
- Fix: Contact details would not allow you to specify Job-Title or Department without specifying Company name.
- Fix: Webkit browser only bug (Safari / Chrome) where the first page of the surgeweb preferences would "sometimes not scroll" - particularly annoying on lower resolution screens.
- Fix: "Resent-From:" header was no longer being correctly populated during redirection - broken some months back.

Tue 2 August 2011: Specials build 5.3i-26

- Fix: Further fixes to format flowed formatting (broken in new and interesting ways in 5.3i-11)

- Fix: Crashing bugfix for corrupt surgeweb cache files (definitely affecting solaris64, possibly other platforms).

Tue 26 July 2011: Specials build 5.3i-24

- New: Setting to "always display attached emails as inline" - particularly useful for receiving multipart-digest messages. (under per user advanced surgeweb settings)
- New: "Send as new" in more actions menu item which allows you to use any existing message (sent folder or any other folder) as a starting point to composing another brand new message.
- New: Setting to enable "resend without reply formatting" to get resend working more like other clients such as Thunderbird. (under per user advanced surgeweb settings)
- Fix: Minor tweaks to display of "redirect" message composition.

Tue 19 July 2011: Specials build 5.3i-22

- Fix: In left list on contacts page "Favorites", "All Contacts", "All Addresses" can now be internationalised.
- Fix: Toggling star by clicking now fully works in widescreen mode, message selection using white space under checkbox tweaked for widescreen mode. Setting flags in ajax interface feels more responsive (ui updated on start of request rather than completion of request)

Mon 11 July 2011: Release build 5.3h2-2 (patched)

- Fix: Several sporadic crash issues resolved

Mon 11 July 2011: Specials build 5.3i-15

- Fix: Reduced excessive emsg logging (introduced 5.3i-11)
- Fix: Several sporadic crash issues resolved

Tue 6 July 2011: Specials build 5.3i-11

- New: Lesser used html editor buttons collapsed to a menu button to make space for several new feature buttons to be enabled soon.
- **New:** Support for iOS devices (iPhone / iPad etc) for the ajax and html interfaces. Not optimised for fat finger touching on touch screens, but at least perfectly usable now with a bit of zooming and scrolling, which it has not been to date.
- **New:** True plain text composition support for ajax and html interfaces (used automatically by iOS devices)
- Fix: Several fixes with the format flowed message text part reformatting.
- Fix: Added viewport meta tag to mobile interface to (hopefully) make it look better.

Mon 27 June 2011: Release build 5.3h-1 (Existing beta distribution upgraded to release)

Fri 10 June 2011: Beta build 5.3h-1

- Fix: Few minor fixes

Tue 24 May 2011: Specials build 5.3g-3

- New: IMAP namespace support for configurations where surgeweb is configured to talk to non surgemail imap servers. To configure a namespace add config_*.ini setting of say "imap_namespace INBOX."
- Fix: Tweaks to handling of mobile login page. If you use the url <http://.../surgeweb?mobile=true> to get to mobile login page, remember me works on the mobile login page too.

Thu 19 May 2011: Beta build 5.3f-1

- Fix: Make surgeweb display correct message date/time for browser detection of non integral timezones
- Fix: Further washing of contacts email field to prevent unexpected email address formatting and prevent things like tab characters doing bad things to users' addressbooks.

Fri 13 May 2011: Specials build 5.3e-2

- **New:** Changes for significant surgeweb webserver performance enhancements should now be fully enabled
- Fix: Forwarded messages with multipart related content were no longer displaying their mpr images properly inline even though the images were attached (broken 5.2d-8 I believe)
- Fix: Bunch of minor mobile template fixes

Tue 10 May 2011: Specials build 5.3d-6

- New: (non surgeweb) Application wide memory handling changes to try and boost performance - disabled by default still. Build should be

fully stable though as far as I know.

- New: Mobile interface now honours the https_required setting too.
- Fix: (non surgeweb) IMAP message date / time (as displayed by surgeweb) maintains message timestamp for mailbox import from gmail.
- Fix: Flash attachments uploader now works under IE9
- Fix: CSV address import would fail if import had a single quote character on any line other than first of a multiline field (eg address field).

Wed 27 April 2011: Specials build 5.3c-29

- New: Additional surgeweb status logging and work done to identify performance bottlenecks on some large servers.

Tue 26 April 2011: Specials build 5.3c-26

- New: Surgeweb config_*.dat setting to disable browser input field autocompletion on the login page (not recommended but may be needed to pass security audits) "disable_login_autocomplete true"
- Fix: Bunch of minor fixes & formatting changes of the basic HTML & mobile interfaces.
- Fix: (Non Surgeweb) Archive extract maintains message time which means unarchived messages get displayed with the correct message time in surgeweb.

Thu 21 April 2011: Specials build 5.3c-24

- New: Mechanism for arbitrary mobile or html template customisation without having to make changes to the tpl directory. Not recommended for customisation unless you really need to, as on upgrades you will need to be comparing your template versions against the ones in the tpl directory to see if NetWin has made any changes and then incorporate any changes. But it is still better than having to modify the files in the tpl directory directly.
- Fix: Minor fixes to single signon url customisation feature added in 5.3c-19.

Wed 20 April 2011: Specials build 5.3c-22

- New: Config_*.dat setting of "help_add_domain true" which will add "?domain=thedomain.com" to all help web requests allowing your webserver to do customisation based on domain without needing to maintain multiple documentation trees per domain.
- Fix: Web encode login_title field of surgeweb customisation page of admin interface, other fields were already web encoded. This allows you to set say " " to blank out a field.
- Fix: Admin interface fix where "custom/" was being repeatedly added to the login_css settings under certain conditions.
- Fix: Variety of other little fixes.

Mon 18 April 2011: Specials build 5.3c-21

- New: Html interface now also does "panelling" borders (no rounded corners)
- Fix: Html interface Selecting contacts and then "doing things" was incorrectly resetting the menu highlight to "mail" again.
- Fix: Html (Basic) interface was listing the multivalue special folders as specified in config files eg "Sent,Sent Items,Sent Messages" instead of "Sent"
- Fix: Surgeweb preferences page was showing an error and not populating user.cgi based info if you had surgeweb ssl set to being required, and the server was not listening on 127.0.0.1 for as specified by g_webmail_secure_port. Now uses first fully specified ip / host it comes across in g_webmail_secure_port.

Sun 17 April 2011: Specials build 5.3c-19

- Fix: Several fixes to yesterdays customisation changes. Also now by passing domain in on command line (eg http://myserver/surgeweb?domain=domain2.com) supports all logged out "login page customisation" - still requires login_crossdomain to be globally enabled.

Sat 16 April 2011: Specials build 5.3c-15

- **New:** Support for surgeweb domain customisation for use with multiple domains on a single sign-on url. Obviously only the "logged in" customisation settings will get applied per domain and login_crossdomain needs to be globally enabled.

Fri 8 April 2011: Specials build 5.3c-12

- Fix: Support for global login.htm file for use with custom_login.htm. ie surgeweb/custom/login.htm as well as the already existing surgeweb/custom/{domain}/login.htm
- Fix: Washing of certain badly formatted spam was taking up way too much cpu.
- Fix: Allow passwords to start with '(' as first character logging in to surgeweb.
- Fix: Ampersand fix, attempt 2 - forgot to add fix to source control for 5.3c-10 :-(
- Fix: Inline display of attached messages (and probably images too) was not working for messages with certain mime constructs. (probably

broken 5.3c-8)

Mon 4 April 2011: Specials build 5.3c-10

- Fix: Support for the ampersand character as part of the username when logging in to surgeweb.

Wed 30 March 2011: Specials build 5.3c-9

- Fix: Message drag and drop into folders did not work in FF4.
- Change: (non surgeweb) On windows we will now be building with newer version of developer studio, let us know if it causes any problems.

Mon 28 March 2011: Specials build 5.3c-8

- New: Support for multipart/digest messages. Note: watch for any odd MIME messages just in case they now display badly. A side effect of this addition is that messages with multiple plain text parts - very rare, but sometimes result of virus scanner modification etc - should now show all parts rather than just the first part as they did previously.

Mon 14 March 2011: Specials build 5.3c-5

- Fix: If you are connected to surgeweb using https (using "https_required true"), proxied user.cgi requests (to complete the "Preferences - Filtering and spam control" settings page spam settings) will be https as well now - allows total lock down of http port. (if needed restore old behavior using "old_usercgi_proxying true")

Fri 11 March 2011: Beta build 5.3b2-2

- Fix: Same crashing bug fixed as fixed in 5.3c-4

Fri 11 March 2011: Specials build 5.3c-4

- Fix: Crashing bug when sending messages from html interface with blank bodies (introduced 5.2d-8)

Tue 8 March 2011: Specials build 5.3c-3

- Fix: In message display total attachments count was getting listed as "N Attached Images" so message with say "image + vcf file" was listed as having "2 Attached Images".
- Fix: Disabled check for "Old draft could not be removed" when sending. Some people were sporadically seeing these in surgeweb leading to confusion, now logs in mail.err instead. Contact support if you see significant numbers of these log entries, as source of issue has not been resolved.

Mon 7 March 2011: Beta build 5.3b-1

- Fix: Pressing Next / Previous on surgeweb's user.cgi blacklist settings page listed friends (whitelist) rather than blacklist addresses (although previously deletion attempts would result in no action and correct blacklist addresses getting displayed again - confusing to users though).
- Fix: Surgeweb's user.cgi whitelist settings were not displaying nicely in IE9 (don't think other browsers affected, not sure about earlier versions of IE)

Thu 3 March 2011: Specials build 5.2d-9

- Fix: All messages sent using surgeweb were being sent without a subject (was broken yesterday)

Wed 2 March 2011: Specials build 5.2d-8

- New: Surgeweb reply to surgevault message automatically enables surgevault on reply message (disable behaviour using "surgevault_reply false" in surgeweb config_*.dat files)
- Warning: Some major surgeweb code restructuring done relating to message part handling. Let me know if any changes are noticed wrt display of various message parts or when replying / forwarding multipart messages, attachments etc (there should in theory be no change in behaviour for now).

Mon 28 February 2011: Specials build 5.2d-5

- New: (non surgeweb) Old webmail support to get users' real ip address in delivery logs during smtp sends (requires g_webmail_secret and smtp_realip settings)
- New: Ability to show arbitrary right column info / iframe on login page (much like the right column in ajax interface). Contact surgemail-support if you want further instructions to use this.
- Fix: Chrome version 10+ (still prerelease) was doing odd page layout things under a variety of conditions when replying to a message.

- Fix: Urls in messages were getting truncated at 200 characters
- Fix: Attachments with a certain slightly odd mime type were not getting displayed by surgeweb

Tue 1 February 2011: Beta build 5.2c-1

- Fix: Two CSV contact import fixes: If individual fields in a csv record were over 10KB this would result in failed import and 100% cpu usage, multiline values in the "Notes" field now get imported into the "Comments" field (up to a maximum limit of 1KB).
- Fix: Copy and pasting multiple addresses at once into to / cc / bcc fields was not getting handled as nicely as it might have been.
- Fix: If an account's quota was set to "(no limit)", the body search and clicking green info icon did not work.
- Fix: Dragging a dropping a contact from shared addressbook to personal addressbook with single quote in name was failing. (previously was working by first drag & drop copy to group without quote, and then from there to group with quote)
- Fix: Pressing Ctl-alt-del (to lock screen etc) with surgeweb browser window as front window was deleting the message selected by cursor.

Tue 11 January 2011: Release build 5.2a-1 (Existing beta distribution upgraded to release)

Fri 7 January 2011: Specials build 5.2b-6

- Fix: If you selected message via click in message list which was the result of a search across multiple folders, then used arrow keys selection cursor would start from the top of list. Now cursor moves from item you clicked.
- Fix: If you clicked a message which was not cached but it took a while to get the message from the server, and you then moved to a next message (by clicking it or arrow keys) that was already cached in browser, the second message would get instantly displayed, but the display would change back to the first message when the first web request returned. Now preview stays on the next selected item you had already moved on to.
- Fix: The "inline display" of attachments / multipart related content from "attached messages" that themselves have attachments or multipart related content does not work. Now this shows a loud warning to use "store in current folder" feature to access original message - and the "non functional" links from nested message content (like show images / slideshow etc) no longer get displayed.
- Fix: Flash attachment fails with error "Flash Uploader - error -220" on https connections with untrusted certificates, now displays a more sensible warning with suggested remedial actions (affects all non IE browsers I tested).
- Fix: Orphaned IMAP folder shares (imap folders shared with other users using surgeweb and then folder deleted without removing share) no longer get displayed in the account the folder was shared with.
- Fix: User.cgi account creation autologin (and "webmail link") now goes to surgeweb rather than webmail. (can restore old behaviour using g_old_webmail_links)

Tue 21 December 2010: Beta build 5.2a-1 (new latest beta build)

Mon 13 December 2010: Specials build 5.1g-2

- Fix: (non surgeweb) Some significant spam handling fixes / improvements.
- Fix: Some of the attachment handling links in messages were not working if the name of the folder that contained them, had a single quote in it.
- Fix: Attachment removal crosses were not working if you did not have Flash installed and were using form based uploads or drag & drop based uploads.

Web 8 December 2010: Specials build 5.1f-2

- Fix: Crashing issue in contacts syntax checking of records >1KB (introduced 5.1d-4).

Tue 7 December 2010: Release build 5.1c2-2 (patched)

- Fix: (non surgeweb) IMAP issue and mail delivery memory leak fixed (same fix as in 5.1e-1).

Tue 7 December 2010: Specials build 5.1e-1

- Fix: (non surgeweb) IMAP issue and mail delivery memory leak fixed.
- New: Make surgeweb send on port 587 instead of port 25 automatically if g_smtp_portforce is set (rather than having to set surgeweb_backend_smtp per domain).

Tue 30 November 2010: Specials build 5.1d-8

- New: (non surgeweb) User.cgi proxying support (g_proxy_usercgi) for use in a proxied server config where the backend server(s) are intentionally not accessible from outside the DMZ.
- Fix: For some oddly formatted html messages (as generated by OSX Mail) the wrong html part was being selected for display.

Fri 26 November 2010: Specials build 5.1d-6

- Fix: Strict contacts syntax validity checking now makes backup of old file before removing any entries, and it was removing entries in two cases where it should not have been.

Fri 26 November 2010: Specials build 5.1d-4

- New: Strict syntax validity checking added to surgeweb contacts handling which should detect any of the past past or possible future single quote formatting issues in the contacts data. Todate these issues have resulted in red javascript error needing manual intervention by admin. Now "bad entries" that do not match expected syntax will automatically be removed (and this action logged to mail.err).
- Fix: Further crash fix related to long strings of invalid UTF8 characters.
- Fix: Html washing made a little stricter to prevent some interactions between DOM elements in message and ajax interface. Let me know if any html messages do not show correctly that used to show correctly.
- Fix: Content of voicemail messages (of non RFC type "multipart/Voice-Message") was getting displayed as text, now displayed as wav attachment.

Tue 23 November 2010: Release build 5.1c-1 (Existing beta distribution upgraded to release)

Mon 22 November 2010: Specials build 5.1d-2

- Fix: Some attachment related links (eg download) were not working if you displayed a message using a multifolder search and the message was located in another folder than was displayed when the search was started.
- Fix: Code to detect absence of Flash and downgrade to form based attachments was not working in IE8.
- Fix: Crash related to long strings of invalid UTF8 characters in base64 encoded messages.

Tue 16 November 2010: Beta build 5.1c-1

- Fix: Surgeweb labels related crash issue
- Fix: Safer handling of really long to addresses containing crap formatting (I think I have a crash fixed)
- Fix: Surgemail image resizing (used by surgeweb) was recently serialised. This was causing restarts sometimes. Let us know if you see any oddities in surgeweb wrt image resize or icons being slow or not happening.
- Fix: Several other minor issues fixed.

Sat 13 November 2010: Beta build 5.1a2-2 (patched)

- Fix: IMAP issue that mean OE6 would not display inbox when adding new accounts (introduced 5.0l-3, outlook and other clients tested do not seem to be affected)

Thu 11 November 2010: Beta build 5.1a-1

- Fix: Memory corruption bug when creating a new label directly from the labels menu that could result in random crashes.
- Fix: When saving spam settings from surgeweb, the resend frequency of html status email was getting reset to whatever it was prior to surgemail version 3.7 (for newer installs that would reset to never send)

Mon 8 November 2010: Specials build 5.0n-15

- New: Automatic throttling and smarter handling of "new mail" checks to reduce performance impact on large systems.
- New: Download page for customer contributed [surgeweb translations](#).
- Fix: Memory leak during background folder downloads from imap (only affects first ever access to folders with more than 200 messages)
- Fix: Recently added skins in custom directory feature were not yet getting applied to html interface (the skins in tpl folder have always worked in html interface)

Tue 2 November 2010: Specials build 5.0n-10

- Fix: Filehandle leak in the message compose attachment upload handling of the html interface.
- Fix: If name contains commas and was not quoted eg *Smith, John* (as opposed to *John Smith* or "*Smith, John*") From header formatting was dodgy with result that from some mail clients you could not hit reply on this message.
- Fix: "Remember me" always went to ajax interface, now goes to whatever interface you actually logged into with "remember me" ticked.
- Fix: "Remember me" did not work with g_url_alias / url_alias base url of http://mydomain.com/ (as opposed to http://mydomain.com/surgeweb which did work already)
- Fix: If logged in automatically as a result of using "remember me" some functions in ajax interface would fail resulting in "probably logged out" warning. (notable examples print, slideshow, drag & drop attachment addition, but others too)

Tue 19 October 2010: Specials build 5.0n-6

- New: Surgemail.ini setting g_host_redirect for use with surgeweb's https_required setting, allows eg. if ssl cert is for https:\\netwinsite.com connections on http:\\netwin.co.nz to get redirected to https:\\netwinsite.com for surgeweb login
- Fix: Was crashing during save draft if using case insensitive filesystem (Windows) and the drafts folder was named "drafts" and the surgeweb settings were set to "Drafts".

Tue 19 October 2010: Beta build 5.0m-1

- Fix: Download of attachments was not working from "additional accounts" (links at the top since 5.0l-4 / individual image links at bottom had never worked)
- Note: Snippets still globally disabled, should be getting enabled in latest builds in near future.

Tue 19 October 2010: Specials build 5.0l-6

- Fix: Multiple accounts bug that would under rare circumstances mix up the use of indexes between accounts resulting in the unnecessary regeneration of the surgeweb indexes from imap.
- Fix: Crash in snippets handling
- Fix: iframe based surgeweb pages (notably surgeplus calendar, filesharing and blogs) would no longer display (bug introduced in 5.0l-4)

Fri 15 October 2010: Specials build 5.0l-4

- **New:** (All browsers other than IE affected) Download of an attachment, save of message in source format, or full page reload/switch to html caused all active background ajax requests to get terminated. At very least, this would result in red unknown error message if there was a background request underway, but if near the start of the session could also result in contacts not working for this session or the inbox caching not working for this session.
- New: For now snippets generally disabled, contact surgeweb-support@netwinsite.com if you want to help test - feature is mostly functional but needs some polish (performance issue, odd behaviour and source of sporadic crashes needing to be resolved).
- **New:** MUCH nicer widescreen view (multiline message display in list, resizable message list width, body snippet support, changed menubar layout)
- **New:** Message body snippet support - displays first text of message body in message list. Currently is only generated when surgeweb gets info from imap to add message to its cache and a display mode that makes use of this is enabled (eg standard widescreen or normal list with display always) - exact nature of this may still be tweaked. Snippet generation and display can be globally disabled by adding "snippets_disable true" to config_*.dat.
- New: Selection handling keyboard modifiers info displayed for multiple message selection.
- New: Reduced browserside inbox caching from a max of 100 messages to max of 20 messages. Means most people still have advantages for most messages, but people that just leave messages in inbox do not cause excessive server loading and disk use. Can still manually cache all on page if really needed.

Tue 12 October 2010: Specials build 5.0j-10

- New: Another sample login screen added with pretty graphics and rounded corners. (thanks to loyal customer for passing that to us for inclusion in surgeweb)
- **New:** Custom surgeweb skins can now be placed in the surgeweb/custom/skins directory and selected from the automatically populated skins dropdown in the admin interface.
- New: Can specify completely custom login.htm file. To do this, set "Custom Login page" to true in admin interface and make sure you create a surgeweb/custom/{domain}/login.htm file.
- **New:** Login screen can now contain "Forgot password", "Create New Account" and custom "Help" links. Enable as [described](#).
- Fix: Custom login page css file setting needs to be prefixed by "custom/" to find the file in the surgeweb/custom/{domain} directory, This now gets added automatically.

Fri 8 October 2010: Specials build 5.0j-7

- New: Support surgeweb global "allow images" addresses by creating a surgeweb/custom/allow_images.dat file with addresses in same format as allow_images.dat in users' folders.
- Fix: (non surgeweb, windows only) extra webmail.exe bughunting logging (enabled with "bughunt_unlock_crashes true" in webmail.ini)

Fri 8 October 2010: Specials build 5.0j-6

- Fix: Oddball case where Webmail distribution list import could result in unusable surgeweb addressbook.
- Fix: First ever login surgeweb temp directory creation fix, with two apparently unrelated symptoms. 1) Webmail distribution lists would not get automatically imported along with the automatic webmail contacts import. 2) Logging in to a system with no surgeweb folder for the user

(notably a mirror system, or frontend system) would reset the users surgeweb preferences (user.cgi settings not affected).

- Fix: Import of some webmail addressbook group names (feature added in build 5.0j-2) would appear !encoded eg "My!20Book"
- Fix: Some odd cases during import of webmail distribution list defined using nicknames, the nicknames would not get resolved resulting in addresses defined as user@local.domain instead of user@original.domain.

Fri 1 October 2010: Specials build 5.0j-5

- Fix: Further fix to date handling. Changes in 5.0j-2 meant that surgeweb was displaying the message date/time out by up to two hours in some cases.

Tue 20 September 2010: Specials build 5.0j-2

- New: When doing imports for any old webmail addressbook other than "default" a surgeweb group now gets created that has all contacts that were part of this addressbook.
- New: Ability to increase number of address autocompletion items. Manually add setting to config*.dat files eg. autocomplete_max 15
- Fix: Messages with a very large number of to addresses containing at least one single quote was resulting in a sent folder that could not be displayed.
- Fix: Improved surgeweb's robustness in dealing with several different forms of unexpected formatting in import of webmail addressbooks and text lists of addresses.
- Fix: Date formatting issue which would get message date/time wrong if surgeweb and surgemail servers were in different timezones (frontend-backend config), or messages were drag and drop uploaded from a server in a different timezone.

Thu 9 September 2010: Release build 5.0e4-4 (patched)

- Fix: Crashing issues below also fixed in production release

Thu 9 September 2010: Beta build 5.0i-1

- Fix: Two sporadic crashing issues

Wed 8 September 2010: Specials build 5.0h-11

- Fix: Several other minor issues resolved.
- Fix: In Firefox message drag & drop was a bit sensitive to "selecting the next message up / down" (FF only)
- Fix: Improved login page browser compatibility test to allow newer versions of existing browsers, notably allows FF4 beta but also covers other browsers.

Mon 30 August 2010: Specials build 5.0h-8

- Fix: Surgeweb preferences javascript error (related to user.cgi) when logging in as a domain admin's accounts that was not an admin of the domain that the account is part of. (eg logging in as user@domain.com and being admin of domain2.com but not domain.com)
- Fix: Made surgeweb "all actions" commands more rfc compliant (ie work again) now that imap is more rfc compliant (as of build 5.0h-5)

Mon 23 August 2010: Specials build 5.0g3-3 (patched)

- Fix: (non surgeweb) Old webmail attachments issue (introduced with 5.0g-1)
- Fix: (non surgeweb) Confusing imap error response issue

Mon 16 August 2010: Specials build 5.0g2-2 (patched)

- Fix: Surgeweb crashing bug introduced with 5.0g-1.

Mon 16 August 2010: Release build 5.0e3-3 (Existing beta distribution upgraded to release)

Fri 13 August 2010: Specials build 5.0g-1

- Fix: Display within surgeweb of user.cgi "disable responder after x days" was not working (it was saving and applying any value you set here though).
- Fix: Multilanguage utf8 translations in some languages were not working in popup message windows under safari.
- Fix: Minor further tweak to the message selection & preview handling.

Tue 10 August 2010: Beta build 5.0e3-3 (patched)

- Fix: 2 further sporadic crashing fixes already resolved in latest development tree

Thu 5 Aug 2010: Specials build 5.0f-4

- Fix: When selecting using checkboxes to prevent message preview / download now explicitly clears preview pane to avoid confusion between messages.

Fri 16 July 2010: Beta build 5.0e2-2 (patched)

- Fix: 2 sporadic crashing fixes already resolved in latest development tree
- Fix: Minor surgeweb template issues already resolved in latest development tree

Wed 14 July 2010: Specials build 5.0f-3

- New: Can now search for messages with some or no attachments - added "With Attachments" to Display menu.
- Fix: Crash when trying to read messages with certain invalid mime / html formatting.

*** <ftp://netwinsite.com/pub/surgemail/specials/surgeweb> no longer in use, latest builds are now in <ftp://netwinsite.com/pub/surgemail/specials> again ***

Fri 2 July 2010: Beta build 5.0e-1

- **New:** (non surgeweb) "Friends pending" IMAP folder name is now "Spam" by default to match surgeweb name, existing configurations remain unchanged but this (g_friends_pending_name "Spam") is now a recommended setting. Surgeweb now also uses g_friends_pending_name and no longer needs imap_spam_folder setting in config_*.dat files. Note existing IMAP copy/move behaviour remains unchanged: IMAP move/copy Inbox->Spam = iss spam training; IMAP move/copy Spam->Inbox = notspam training.
- **Fix:** OSX only fix for longstanding issue of "some of many concurrent requests" failing. Affects all protocols on all ports but two particularly noticeable cases: Connecting to surgeweb under HTTPS in Safari on OSX - a few images and other parts of the interface randomly fail, Connecting to admin interface does not show some graphs.
- Fix: (non surgeweb) Solaris & osx platforms were crashing during every email processed on smtp due to bug introduced a few days back.
- Fix: Several cases of the attachments remove links were not working.
- Fix: On certain Nokia phone formatted messages with attachments surgemail would crash when viewed from surgeweb (introduced with fixes in 4.3k-4)

Thu 1 July 2010: Beta build 5.0c-1 (failed internal testing, not made public)

- Fix: Removed the Attach from Filestore link (until implementation is complete)
- Fix: Removed prototyping code added to the html editor link button on the html editor which made it non functional (will be restored once fancy link addition and image addition code is complete)
- Fix: Failed surgeweb logins were not getting logged to login_failed.log (although g_bad_login_* / g_honeypot_* limits were already getting applied)
- Fix: Crash during send on some particularly unusual formatting in recipient addresses.

Mon 28 June 2010: Specials build 5.0b-6

- **New:** Rewrite of the attachments handling when forwarding messages. Should be much more reliable in terms of not losing attachments when sending. New features include: auto scaling of wide images to fit message window, being able to forward images in messages with "inline" (as opposed to MPR) images (notably as sent by iphone), slide show of attached images, ability to remove individual mpr images when forwarding, ability to treat mpr attachments as normal attachments.
- New: Improvements to caching of images that attached to a message so display images do not have to be redownloaded if a message is forwarded (or in some cases displayed multiple times).
- New: Link added to permanently hide the information line telling you that you can use desktop drag and drop
- **Fix:** HUGE speed improvement in the sending speed of messages with large attachments. (it is possible this will even noticeably reduce loading on large systems)
- Fix: Attachment download links and multipart related images did not work (Chrome 6.0 only)

Fri 25 June 2010: Specials build 5.0a-1

- New: Version number change and full set of specials builds for all platforms in preparation for a new release.

Fri 25 June 2010: Release build 4.3g4-4 (patched release build)

- Fix: (non surgeweb) Domains with mixed case domains in surgemail.ini were no longer able to login (fixed in 4.3h-30 in development branch, unsure when introduced)

Thu 24 June 2010: Specials build 4.3k-16

- New: Spam/Notspam/Allow/Block buttons on likely spam now work when full message is displayed. (previously only worked from preview window)
- Fix: Previous / Next buttons were no longer working in popup message windows if the preview window was not displayed at all.
- Fix: Search for conversation (speech bubble button) was not working anymore in some cases.
- Fix: Several further fixes wrt attachments handling rewrite that were not working quite right yet.
- Fix: Desktop drag & drop attachments were not working in popup message windows yet.

Wed 23 June 2010: Specials build 4.3k-13

- Fix: Enabled the desktop drag and drop for Fluid under osx.
- Fix: Most of the attach remove options were no longer working in 4.3k-12, now most work again. Still investigating how to fix some minor issues.
- Fix: Flash based uploader has a workaround for IE bug to make it work in IE which means the swf file needs to be redownloaded from server for each message you attach files to. This workaround has now been disabled for all other browsers other than IE.
- Fix: Timing issue that resulted in forward with attachments / forward attach "frequently" failing in build 4.3k-12 fixed.

Tue 22 June 2010: Specials build 4.3k-12

- New: "Prettified" the folder management page.
- **New:** Folder sizes on folder management page. Surgeweb tracks folder sizes of folders accessed from surgeweb and displays this on the folder management page, if changes are made not through surgeweb these are not guaranteed to be uptodate in surgeweb (for performance reasons). However there is a button to recalculate these on the folder management page if needed.
- **New:** Shared folders - the ability to use surgeweb to configure ACL based sharing of IMAP folder between local accounts using (requires g_imap_acl to be set)
- **New:** Desktop integrated drag and drop support for adding of message attachments or uploading message files direct to the displayed mailbox by dragging from the desktop or Explorer / Finder windows. Works in recent versions of Chrome, FF, Safari on both PC and Mac. Will not work in IE as the browser does not support the necessary features (unless Chrome Frame is used in which case it will work in IE6, IE7, IE8). DND attachment uploads will also continue uploading in the background while you view other messages etc (unlike Flash based uploads which stop in this situation).
- **New:** Rewrite of the ajax attachment uploads. Now actively detects whether an uptodate version of Flash is available, and if not it degrades to a simple single file form based upload dialog.
- Fix: Server defaults for read receipt sending changed to "Never | Never | Never" - still customisable by admins or users.

Thu 17 June 2010: Release build 4.3g3-3 (patched release build - only solaris 8 distribution has been rebuilt)

- Fix: Solaris 10 sparc only crash issue fixed as per 4.3k-4

Mon 14 June 2010: Specials version 4.3k-5

- **New:** Full surgeweb ajax support in IE6 through the use of the google ["Chrome Frame" plugin](#). The use of this plugin also offers significant advantages when running IE7 and IE8. This includes some rendering artifacts not appearing, it being much faster, and support for surgeweb desktop drag and drop integration as above.
- New: "Save to file" option to download multiple messages as raw messages / text for saving locally from message list. Multiple messages are downloaded as a single compressed messages.zip file.
- New: Incomplete addresses get expanded by surgeweb to include domain during send to make sure headers are complete etc. (eg. 'joe, john' to 'joe@mydomain.com, john@mydomain.com')
- Fix: You could not reply to messages if iconv translation of utf8 characters in From header failed, now it instead displays undecoded (?=...) but at least you can reply to these messages now.
- Fix: Support for most cases of "name*" encoding of attachment names in mime headers

Sat 6 June 2010: Specials version 4.3k-4

- **New:** (non surgeweb) Major new spam prevention system is now on by default. For more info see: [myrbl help](#).
- New: Tweaks to spam handling via surgeweb. Block button no longer reports to iss spam training, and is available for all messages (wherever there is the Spam button currently) - this should be used to block email when it is not actually spam. Various spam actions (Spam / Block / purge etc) more reversible by placing messages back in Spam folder or in Trash rather than outright deleting them.
- **New:** Actual "Read Receipt" sending support added!! Three tiered system that allows you to configure the sending to "addresses on local domain / addresses on friends list / unknown addresses" each as "automatically / only after asking / never". Defaults configurable in config_*.dat files and customisable per user. System defaults are Local(ask)|Friends(ask)|Unknown(never).

- New: Request confirmation "Both" option added.
- New: Handle Macintosh generated "multipart/appledouble" message types.
- Fix: Solaris 10 sparc only issue (worked fine on solaris8 sparc) where surgemail would crash on every surgeweb login. (broken since changes to make sessions last across surgemail restarts a few months back)
- Fix: Formatting issue in true plain text sending. Would only have affected the mobile template.
- Fix: Without preview pane displayed some cases were marking messages as read when they should not have done.
- Fix: Crash on certain really deep or invalid mime constructs in messages.

Thu 3 June 2010: Release version 4.3g2-2 (patched release build)

- Fix: Friends handling sporadic crashing issue (broken 4.3f-14, fixed 4.3h-30)
- Fix: Bulletin crash issue triggered by certain unusual MIME formatting in manually / email generated bulletins (fixed 4.3h-30)

Wed 26 May 2010: Specials version 4.3j-2

- Fix: Rewrite of the SSL "login only" option to make it compatible with more browsers without showing warnings when switching from https back to http. Also an option added in the users preferences to keep the whole session https (only displayed in preferences if "login_only" option is enabled on surgeweb admin page).
- Fix: SSL "login only" option was broken in 4.3i-1 :-(

Mon 24 May 2010: Beta version 4.3i-1

- No significant surgeweb changes over 4.3h-30

Fri 21 May 2010: Specials version 4.3h-30

- **New:** SSL for "login only" option. This forces ssl for the session login but then switches to http for the rest of the session. (means browser does not display "unsecure" warnings for images etc in messages and slightly improved performance. FYI as an aside: Flash based attachment uploads don't work with self signed SSL certificates, ok on real ssl certificates)
- Fix: Sporadic crash in surgeweb's folder handling.
- Fix: Minor fix on folder management page for multiple language support.
- Fix: Background "first ever" download of messages in large folder (>200 messages) on alternate accounts was not downloading all messages and left the folder in "calculating..." state. Background download now works, and additional fix that "calculating..." times out after 30 minutes. So if any folders are in this state for whatever reason (eg a surgemail crash) these should now autocorrect rather than requiring an manual surgeweb cache reset.
- **Fix:** mail.err / logging tidyup (surgeweb and non surgeweb) to try and track down any unusual surgeweb errors that may have been going mostly unnoticed.
- Fix: Surgeweb was not deleting its cached message body files when messages were deleted or moved to another folder, resulting in surgeweb caches that grew larger than needed - particularly on high volume accounts. (Although the cached message files get automatically removed within a week anyway)
- Fix: Labels management panel was no longer scrolling if you had a large number of labels (not sure, but I think this was only broken recently)
- Fix: Sending messages using surgeweb with really long subject lines containing lots of utf8 characters could crash surgemail.

Thu 13 May 2010: Specials version 4.3h-23

- New: Actual foldernames in the mobile interface were not yet displaying language translated, also foldername displayed correctly translated at top of list in mobile and html interface.
- New: Minor tweaks to the folder management page and made all strings on the folder management page language translatable (both ajax and html).
- Fix: If the whole of surgeweb was embedded in an iframe some of the buttons on surgeweb's iframe based dialogs were not working (eg contacts import / export, or multiple account configuration)
- Fix: Tweaked output headers in mobile interface to make multilanguage UTF-8 characters display better on some browsers (notably Firefox)
- Fix: Few css syntax errors fixed that resulted in browser console warnings.
- Fix: Contacts picker was not working properly if "Search for contact..." had been translated to another language using the language translation table.

Thu 13 May 2010: Specials version 4.3h-22

- **Fix:** Allow you to start typing in search box after clicking folder but before the folder is displayed (previously actual list display switched focus, resulting in keyboard shortcuts getting triggered on n / r / a etc keystrokes)

- Fix: Firefox only issue where "Communications error" would get displayed if an existing foreground web request was cancelled by new foreground web request. eg clicking one folder / message(if message not already in browser cache) and clicking another before first fully displayed.

Mon 10 May 2010: Specials version 4.3h-20

- **Fix:** Existing messages in the mailbox sometimes not getting displayed by surgeweb fixed!! Problem occurred sporadically under a specific set of conditions but generally only on the inbox and had to have more than 100 messages in the inbox (the more you had the more likely it was to occur, previously if it happened a full surgeweb cache reset was required to get the messages back in the message list). Bug fixed threefold (any one of which is actually enough to prevent this happening in the future): core fault fixed that corrupted the index, intermediate code that detected the corrupt index and tried to remedy this fixed, and new code added that will correct any folders found to be in the faulty state. So any accounts that are currently affected by this issue will show any "missing messages" again without additional intervention after installing this build.
- Fix: Switching between multiple accounts suffered from the same bug where reply would sometimes fill in your own address (was resolved for Refresh link in 4.3h-11)
- Fix: If messages were deleted from the inbox before the inbox caching request was actually sent, this would result in a red javascript error, and caching would fail.
- Fix: When doing forward of a message with attachments or a "forward attach", if the web request came back before the flash uploader was initialised this resulted in an error and the attachments failing to get attached. (most noticable for tiny attachments whilst using IE)
- Fix: If no folders under "More folders" had been created, and the last special folder (ie probably Trash) had a subfolder the labels and search display in the folder tree was getting messed up.
- Fix: Admin interface link on the "value added features" page now actually goes to the surgeweb customisation page (previously only the link in the folder tree did).
- Fix: Further improvements to response code handling so that surgeweb displays more sensible error when smtp sends fail for reasons like to many open smtp connections.
- Fix: If a message subject line had multiple "=?..." encoded sections and any other than the first were invalidly formatted this could result in surgemail crash when viewing the message with surgeweb.

Wed 5 May 2010: Specials version 4.3h-17

- Fix: Various inline attachments (like MPR images) were failing to display for the additional accounts.
- Fix: View link next to icons of attached images was not working for additional accounts - works now. (I don't think has ever worked for popup message windows - fixed now too)
- Fix: Pressing Spam / notspam actions within the additional accounts was failing (and reporting an error). Works now.
- Fix: %25 Spellchecker bug. If you ran the serverside spellchecker, and messages contained any % characters they would be turned into %25 in the actually sent message even though the message in the editor still looked correct. Let me know if this change results in any spellchecker errors getting reported.

Mon 3 May 2010: Specials version 4.3h-16

- Fix: Last minute tidyup in 4.3h-15 introduced bug that stopped a reply to an existing message from working. Fixed.

Mon 3 May 2010: Specials version 4.3h-15

- **New:** Send vCard by email
- **New:** Export contacts (as vCard or CSV or list of addresses). Try the export with some real world data sets and let me know if you find any problem - there are bound to still be a few teething issues but it should mostly be fully functional. I do still plan to add the group definitions to the vCard export of all contacts.
- **New:** Import arbitrary "copy-paste-able" list of addresses as part of the contacts import dialog.
- New: Correctly import webmail "groups definitions" that are specified using multiple addresses in a single email address field of a webmail *.abk entry.
- New: Display groups that contacts are "a member of" as part of contact details / contact editing (read only).
- Fix: Changed the wording on the default error when surgeweb failed due to session no longer being valid (previously error suggested the template file was missing), and now still allow surgeweb dialog close in this situation.
- Fix: Filter out Cmd-N so it has standard behaviour on OSX.

Sat 1 May 2010: Specials version 4.3h-14

- Fix: Nasty bug where a download all would remove the surgeweb index files resulting in weird surgeweb error messages afterwards requiring surgeweb relogin, and dependign on user actions causing surgemail crashes as well (introduced 4.3h-10).
- Fix: Use unqueued web requests for the empty message send template and existing attachments retrieval. If message sending is slow (eg a

very large message is getting sent) this means you can actually be replying to the next message rather than after pressing the reply button having to wait till the previous message was sent before the reply interface appears.

Fri 30 April 2010: Specials version 4.3h-13

- **New:** Full customisation of foldernames of the special folders (Inbox, Drafts, Sent etc) for multiple language support. The actual imap foldernames are set on customisation page and should generally be left the default English names and lang_*.dat now have examples how to translate surgeweb display name to any language. (for now the folders management page still displays real imap names but this page will be tidied up soon).
- New: Proxy mode using "tohost" support added. Previously mail and contacts related functionality already worked but user.cgi / surgeplus etc resulted in an error. In proxy mode surgeweb's imap now goes direct to proxy server rather than through local surgemail imap first (for efficiency).
- Fix: Webmail imports were disabled / non functional as a result of additions in 4.3h-10. Fixed.
- Fix: Preemptive caching of new messages that arrive (done for instant display, and only done for the account you currently have selected) was no longer working since the introduction of multiple accounts, and sometimes resulted in a red javascript error.

Thu 29 April 2010: Specials version 4.3h-11

- Fix: Opening a message in tabbed mode then switching folder, then going back to already open message and replying resulted in error. Longstanding bug I believe.
- Fix: Opening a message in a popup window and pressing reply/forward resulted in error message (new in 4.3h-10)
- Fix: Longstanding bug where surgeweb would sporadically fill in your own address when you replied to a message. (finally located, simple when you know where to look!! was the result of switching from Sent folder to Inbox via the Refresh link)
- Fix: Fixes to new_mail checks for multiple account support. Notably, the message list and unread count in folder list would sometimes get incorrectly updated.

Wed 28 April 2010: Specials version 4.3h-10

- New: Reply-to address can be set per account for the base account or any of the additional imap accounts.
- **New:** Multiple local imap account support, to allow you to work on multiple separate imap accounts at the same time whilst logged in to surgeweb. All accounts are monitored for new mail and you can select any of the accounts when composing or replying to messages - this "actually sends" mail from the alternative accounts (in terms of headers, smtp authentication etc). Where relevant the preferences of your base account are used for all accounts and a few features that need your preferences can only be used on the base imap account. (for now still disabled by default but enable using "show_identities true" in surgeweb custom/config*.dat files)
- New: Surgewall support. Contacts, labels, etc should now work in surgewall configuration. The notable missing surgewall feature is that the surgemail level spam folder is not available yet though - I'll try and fix that soon.
- New: History of status & error notifications is available by clicking green status icon. Also there is a button to directly report these via email - saves trying to copy and paste errors that rapidly disappear sometimes.
- New: The one to one mapping from surgeweb browserside to surgeweb serverside to IMAP serverside has been made more flexible to support several new features, notably multiple accounts, multiple identities, and surgewall support. These are all mostly implemented and will be documented here as I test and add each feature to the main build tree.
- Fix: Various fixes to the multiple imap account functionality.
- Fix: Doing an inplace clean of a message subject using "notspam" would update the display with a subject where the ascii string was not converted to utf8 making it display as "=?..encoded.." until the next message list refresh.
- Fix: If a relogin dialog is displayed, newmail checks are stopped. In particular this fixes the issue that sometimes the dialog would get cleared & redisplayed as you were typing in it.
- Fix: Show imap folders with nonexistent parent folders nicely. eg. if an imap folder fld1 and fld2/subfld existed but no fld2, previously this would display subfld as a subfolder of fld1. Now gets displayed correctly. Notably this means that folders shared between accounts (using ACL - needs g_imap_acl) display nicely.
- Fix: Error handling improvements that means that some problems that may have gone unreported before should now get appropriately displayed to the user. Also further improvements in the formatting of the errors that get displayed.
- Fix: Surgeweb work path of host_alias'd logins are now stored with a path based on the main domain.

Fri 16 April 2010: Beta version 4.3g-1

- **Fix:** Improved automatic special folder detection to match other mail clients. If a folder "existed in the past" but had been deleted and was still present as a deleted entry in surgeweb's folder index it would still try to use it as the special folder making it kinda hard to "change surgeweb to match" another mail client if you had logged in using surgeweb before logging in using your other mail client or whilst these other folders existed for some other reason.

Wed 14 April 2010: Specials version 4.3f-21

- Fix: Improved on fix in version 4.3f-7 as html parts were now NOT displaying in some other messages they should have been.

Fri 9 April 2010: Specials version 4.3f-20

- New: For manual new contact creation, whilst a group is selected automatically creates the contact as part of the selected group. Makes this much more usable.
- New: When creating a new contact is smarter about automatically selecting the addressbook / group that the contact is created in and scrolling the address list to show newly created contact.
- New: Option to choose contacts displayed sort order first/last or last/first. Default is "Last name / first name".
- Fix: Surgemail import of webmail distribution lists did not work if user had added "Nicknames" in the distribution list (it added surgeweb contacts with address nickname@local.domain instead). Now converts nicknames to webmail addressbook entries with fullname and email address before adding as surgeweb contacts.
- Fix: Contacts without any first or last name were getting incorrectly displayed unsorted, now uses email address for sorting in this case.

Tue 6 April 2010: Specials version 4.3f-17

- Fix: In some cases the subject washing was failing to remove the stars etc, should do so more reliably now.
- Fix: Crashing bug fixed in spam handling rewrite code.
- Fix: Last minute change to the spam handling rewrite code for a fix on g_maildir_netwin based systems broke most of the actual white and blacklisting. Works again now.

Tue 6 April 2010: Specials version 4.3f-14

- New: (non surgeweb) new setting that allows user.cgi / status email / imap releases / friends bounce releases to be subject washed to remove stars (g_friends_release_wash).
- **Fix:** Complete rewrite of the way messages are marked spam / not spam from surgeweb fixing a bunch of oddities and making the code a lot more maintainable. It now works as it was intended to all along, for more info see full [behavioural description of fix](#) and original [documentation](#).
- Fix: Several minor fixes to the new surgeweb session handling feature introduced in build 4.3f-6.
- Fix: HTML washing fix to better deal with double quotes in inline style string definitions - notably makes some Outlook generated messages display a whole lot nicer.

Fri 26 March 2010: Specials version 4.3f-8

- New: Make multipart/related messages with image identification through Content-Location header instead of CID header work as well.
- New: Display attached email message inline, now reversible with "remove inline display" link. Also some email attachments were not detected as emails.
- New: Allow multiple messages to be processed at once with the "Spam / Not Spam / Allow once / Block forever" buttons on the message list page.
- Fix: When actioning multiple messages as "Spam / Not Spam / Block forever" make sure the messages that end up in the "- Train ..." folders are marked as read to prevent new message indicators in tray to show there being "new messages".
- Fix: CID images with square brackets part of the "cid tag" were not getting displayed inline.
- Fix: Setting "Options - Advanced - message send mode" required page reload / relogin, now applies straight away.

Thu 25 March 2010: Specials version 4.3f-7

- Fix: If no preview was displayed then opening messages in tabs would not use the correct message subject for the tab name.
- Fix: If signature contained escaped characters (eg single quote) an additional forward slash would get added each time you pressed save on the preferences page.
- Fix: multipart/mixed plain text messages with an html attachment were showing the html attachment as the message display part. (ie getting treated as multipart/related)

Mon 22 March 2010: Specials version 4.3f-6

- **New:** Rewrite of the surgeweb session handling. This means sessions will survive surgemail restarts & upgrades, and long browser sleeps. Sessions should by default last up to 2 weeks if you are using surgeweb daily. Idle timeout default is 48 hours. These are both customisable.
- New: As part of the session handling changes, the session id's numbering scheme has been made "less guessable" to avoid possible abuse of this for XSS attacks (was already kind of unlikely but now is safer).
- **New:** As part of the session handling changes, a full record is kept of surgeweb login / logout events with SID + IP address + account information.

Fri 19 March 2010: Specials version 4.3f-3

- Fix: Multiline headers were not correctly decoded which meant that the subject line (or To / Cc headers probably too for that matter) could display non decoded text sometimes eg "=?utf-8?B?..." This does not fix any values already in the surgeweb header cache, manually reset the surgeweb cache to clear any existing bad entries.
- Fix: Fixes on the surgeweb login page and several other places to better protect against XSS attacks.

Mon 15 March 2010: Beta version 4.3e-1

- Fix: Download all can only zip up one set of messages at a time. If it cannot get it's lock in short order it was restarting surgemail now it shows a "try again soon" message to user.
- Fix: Download all was failing for 250+ attachments per message (32 bit machines) and 125+ attachments per message (64 bit machines)

Mon 8 March 2010: Specials version 4.3d-3

- Fix: (non surgeweb) The email that triggers the spam status email does not get listed in the status email (by design) but was getting purged if you pressed the "Purge" or "Spam All" links. Now it remains intact for the next time the status email gets sent.
- Fix: Labels with no ascii characters at all could result in crashes or odd behaviour. This now works for any UTF8 characters, but no attempt is made to try and correct any labels generated with previous "odd behaviour" as there is no easy way to fix these.

Mon 8 March 2010: Beta version 4.3c-1

- Fix: Several non surgeweb fixes (SMTP related crashes in 4.3a-1, DNS lookup bugs in 4.3b-3)

Sat 6 March 2010: Specials version 4.3b-3

- New: Now can rename / move whole folder trees including subfolders
- Fix: It was not possible to move sub folders "out of the folder" they were a subfolder of.
- Fix: Move folder was not fully encoding parameters which meant it did not work with folders with UTF8 characters etc in the name.

Thu 4 March 2010: Beta version 4.3a-1

- Fix: Hide the Email vCard button for now as it is not yet implemented.

Tue 2 March 2010: Specials version 4.2g-33

- New: Moved the Done folder to being a Top level folder by default (can be dropped back to "More folders" using arbitrary top level folders feature)
- New: Support for arbitrary top level folders (and the ability to drop system folders to "More folders" if desired).
- New: Improvements to the Templates folder. Rather than being obscure/hidden there is now an explanation and link to create it from the folders management page. Templates folder can be deleted, which was not possible before.
- Fix: (non surgeweb) several fixes to spam_held to friends_silent conversion code (if g_friends_default_mode was not yet set (to silent or list), friends spam score would get set to zero; accounts with no per user settings stay that way; log file generated; tellmail held2pend_email command to set spam email frequency for all users)
- Fix: Resend now keeps attachments
- Fix: It was not possible to do a forward_attach of an email with multipart related content (broken with changes a week or two back, resulted in a red javascript error when it tried to attach the email). Also tidied naming of forward_attach of messages with blank subject line.

Sat 27 February 2010: Specials version 4.2g-32

- New: (non surgeweb) Tellmail command to convert accounts from spam_held to friends_silent spam quarantening as noted in [recommended spam quarantening](#).
- **New:** Improved contacts organisation using drag and drop. Normal drag and drop still adds contacts to groups, however drag and drop between addressbooks now results in a copy of the contact information getting added to the other addressbook. This is useful for copying from shared addressbook to personal addressbook, and for editing shared addressbooks if you have edit permission. Also allows adding contact to groups in shared addressbooks using drag and drop (as has always been possible with personal contacts).
- New: Pressing Ctl+Alt on login (Ctl+Cmd on Mac) logs in without automatic message marking as read (same as ctl+alt folder selection previously already functional)
- Fix: Minor fix in formatting of login page which was being displayed oddly in IE with certain css formatting.
- Fix: Minor rewording of surgeweb spam handling related features (mostly the user.cgi dialog)
- **Fix:** In Chrome disabled "onbeforeunload" based warning when navigating away from email you are editing, as having these enabled seems to somehow trigger the Chrome browser bug which results in surgeweb frequently becoming totally non responsive using Chrome. Bug

[noted here](#) in google's documentation.

- Fix: Surgeweb logging related oddities when using (g_)url_alias which may _very possibly_ have been causing recent crashes.
- Fix: Fixed shift click handling when forwarding / replying to messages in tabbed mode (FF and Safari only). Notable broken behaviours: shift clicking for selection then doing message actions like forward did a popup when it should not have, shift clicking many actions that should have done a popup did not.

Tue 23 February 2010: Specials version 4.2g-27

- **New:** When sending to multiple recipients try all before reporting failure error rather than failing on the first one. Prepend the error with the recipient address. This makes it a lot easier to find bad recipients when sending to multiple recipients or groups which may contain many addresses.
- New: Surgeweb now applies g_bad_login* checks, and also the new g_honeypot_* checks to avoid password guessing.
- Fix: Select and sort menus were slightly messed up as a result of the language translation additions.
- Fix: (non surgeweb) Fixes so that html based status email get sent out even if just using spam held. For more info see recommended [spam quarantening](#).
- Fix: Print screen did not explicitly have utf8 meta header. Mostly utf8 would display fine due to browser character set detection, but if there was any bad character encoding in the message it would default to ascii instead of utf8. Now will display as utf8 regardless.
- Fix: Focus stealing issue (IE only). When in popup windowing mode, under some circumstances, if reading/editing a message in a popup window, having new messages arrive would drop focus back to the main message list window. The code to center the message list on the old cursor was doing this - only under IE though. Now in this circumstance the "center on old cursor code" is disabled and focus is correctly retained in the message windows.
- Fix: The very last contact in a personal or shared addressbook could not be deleted via the "Edit" contact - "Delete" button.

Thu 18 February 2010: Specials version 4.2g-23

- **New:** Full language translation support across "most" of the surgeweb interface as [documented here](#). Please report to surgemail-support@netwinsite.com any phrases that are no longer displayed correctly or are incorrectly translated.
- New: Significantly speed improvements in the surgmail template parser (used by surgeweb / user.cgi / surgeplus / admin interface, but not old webmail). Template function processing and variable replacement is 2 to 3 times as fast now as it was.

Wed 17 February 2010: Release version 4.2d4-4

- Fix: (non surgeweb) Patch of current production release for crashing bug for certain oddly formatted spam messages, and have help pages correctly formatted.

Sat 13 February 2010: Specials version 4.2g-19

- Fix: Further improvements to attachments handling. Attachments with really long filenames now get correctly displayed, and some attachment problems since the changes in build 4.2g-12 resolved (attachments with unusual formatting in headers were no longer getting displayed, and some text attachments were getting displayed inline instead of as attachments) .

Fri 12 February 2010: Specials version 4.2g-18

- Fix: (Solaris8 64 bit only) Iconv conversion issue that notably resulted in redbox javascript warning in recent builds when going to "Options - Preferences". May well have affected other iconv usages (such as display of messages requiring character set conversion etc).

Fri 12 February 2010: Specials version 4.2g-17

- **New:** Support full UTF8 based searching both surgeweb index based and IMAP body searches. This required some subtle changes to surgeweb indexes, and these will get recreated from imap after installing this updates (must click a folder to "display it once" before searching using surgeweb indexes).
- New: Made surgemail mbox raw message file attachments get handled the same way as attached email messages.
- Fix: Crash if really long links were encountered in messages.
- Fix: When doing a forward attach of other messages limit the filename (based on the message subject) to 40 characters.
- Fix: Tweaked the "new mail" star indicator to have lower priority than the replied / forwarded indicator.

Thu 4 February 2010: Specials version 4.2g-14

- Fix: Make the mobile interface logo customisation work on the mobile interface login page as well as the actual mobile interface pages (for more info [see help](#)).
- Fix: Domain shared addressbook caching mixup. If you used Domain addressbooks on multiple vdomains, surgeweb would try and cache these in the same file resulting that the last addressbook to be changed was getting displayed to all vdomains. (introduced 4.2e-4)

- Fix: Non surgeweb - further bug imap bug introduced in 4.2g-12, which was resulting in surgeweb any message "moved to a folder" getting displayed with a "new message" star indicator.

Wed 3 February 2010: Specials version 4.2g-13

- Fix: Messages with long complicated javascript in the html parts were not getting correctly washed.
- Fix: Non surgeweb - two bugs introduced in yesterday's 4.2g-12 build (extra debug logging was slowing surgemail too much, and some strange imap message handling behaviours wrt read status of messages and messages "not deleting")

Tue 2 February 2010: Specials version 4.2g-12

- **New:** Support for messages with interleaved inline display parts. Notably some Apple devices and software send messages this way. Previously only the first text part was getting displayed and the images were getting displayed as normal attachments. Now all text parts get displayed and the images are displayed inline. Note: old attachment handling behaviour can be restored with "Preferences - advanced - Disable inline images", although this does display the additional text parts which never used to be displayed.
- **New:** Can now handle attached email messages from within surgeweb. Two options are presented when attached email messages are detected: 1) display the attached emails inline in the current message (allowing you to read / reply etc), or 2) store the attached emails as self-contained messages in the current folder (allowing you to reply / file etc as appropriate).
- **New:** Better handling of reply / forwarding images in multipart related messages. If forwarded all multipart related content (ie generally images) gets forwarded as well (previously the images were not forwarded, and a warning to this effect was displayed to the user). When replying to messages containing mpr content, the images are replaced with the text "[Image]", rather than displaying broken links as it did previously.
- Fix: Really long subject support. If the subject was longer than a few hundred characters message send / save draft etc failed to work, fixed now.
- Fix: Sending to an email address with single quotes in the email address to text description by replying to a message resulted in an invalid contacts entry getting saved. This now creates a valid correctly encoded entry.

Fri 29 January 2010: Specials version 4.2g-7

- New: Rewrite of the way surgeweb deals with attachments and what to display from a message to support multipart related message sending, display of attached rfc822 / *.eml messages allowing viewing / replying to in surgeweb, and fancier mime formats that interleave multiple MIME parts inline. None of these features are actually enabled yet, but I will carefully test and enable these one by one over the next few days. Other than aforementioned features I think existing behaviour should stay the same. If you have any messages that used to display ok that no longer display correctly please notify surgemail-support@netwinsite.com.
- New: Allow any text to be customised when using english interface using language translation mechanism eg \$\$\$Some text\$\$\$. Add "custom_english_phrases true" setting to config_*.dat and then add appropriate "English:" entries to the lang_web.dat file.
- New: Resize support for icons at bottom of message for image attachments with IMAGE/JPG mime type, previously it only resized jpg images with IMAGE/JPEG or IMAGE/PJPG mime types.
- New: Changes to attachment handling so images that are part of multipart related messages but have no CID / "Content-ID" header get displayed as normal attachments.
- New: Display multipart/report emails correctly.
- Fix: If encoded (eg quoted printable) plain text parts contained naked CR characters the rest of the line was getting truncated when displaying the message in surgeweb. This now gets corrected and the rest of the line displayed.
- Fix: Keep spaces at start of lines. During plaintext to html conversion space characters at start of lines were lost. This would affect two main circumstances: 1) Viewing messages formatted in plaintext (either received emails or as sent using plain text mode) 2) Saving a message as draft (whilst in plaintext sending mode) and then reediting the message would lose space at start of lines.
- Fix: As part of the daily surgeweb directory expiry, actually delete bad attachments.zip as left by the bug fixed below, to reclaim the disk space :-)
- Fix: A certain case of dealing with "download all" of the attachments would fill your disk with a single large multi gigabyte "attachments.zip" file until surgemail noticed it was too busy and restarted; or your disk was full :-)

Thu 28 January 2010: Release version 4.2d3-3

- Fix: Get the "download all fix" noted above into the production release distributions. Possible signs of this issue having been hit: Major jumps in disk usage by the surgeweb folder, restart reports in startstop.log referring to MZIP mutex. Note: Subsequent to this build (4.2d3-3) it was noted that if this issue is hit, these temp folders files are not actually getting deleted after a few days as I thought they were, this is fixed in builds 4.2g-6+ (or if this space needs to be recovered more urgently manually delete large attachments.zip files in the surgemail/surgeweb/work tree).

Wed 20 January 2010: Specials version 4.2g-2

- New: Print shows all recipients in a message rather than limiting each to/cc line to a maximum of 3 with "...xN more".

Wed 20 January 2010: Specials version 4.2g-1

- **New**: Surgeweb stores Bcc information with messages saved to the "Sent" and "Drafts" folders. This is shown when these messages are viewed using surgeweb (or other mail clients) and gets filled out when you are re-editing a draft message.

Fri 15 January 2010: Beta version 4.2f-1

- Fix: Deal with dragging message list / preview resizer to the very top of the screen. Message list is now always a minimum of two message lines.
- Fix: If ',' or ';' characters are entered in the Firstname or Lastname fields addressing when sending works correctly now.
- Fix: Replying to addresses formatted with doublequotes and containing a comma (eg "Bloggs, Joe") was sending these as doubly doublequoted eg (eg ""Bloggs, Joe") which showed recipient addresses incorrectly displayed in the mail clients of the person receiving the email.
- Fix: Further fixes to UIDVALIDITY resets. I think the odd crashes some people have been seeing since just before Christmas are finally resolved now.
- Fix: Crash if you got two identical logout requests on the same surgeweb session and you were very unlucky with your timing.

Sat 9 January 2010: Specials version 4.2e-37

- Fix: Yesterdays attachments fix was a bit too severe, messages with attachments and message bodies was not displaying the message bodies at all anymore. Fixed now.

Fri 8 January 2010: Specials version 4.2e-36

- Fix: Multiplart/mixed messages with an attachment but no message body were displaying the attachment content as the message body, and not displaying the attachment.
- Fix: Message send request in Ajax interface had slightly invalid http formatting. This was fine in the browser and surgemail, but in the case of one customer network level http request washing was "eating" these so all surgeweb message sends would fail with "Communication error - server not reachable".

Thu 7 January 2010: Specials version 4.2e-35

- Fix: Several memory leaks surrounding surgeweb contacts handling, webmail contacts import into surgeweb, and some IMAP actions.

*** above here particularly notable features or fixes are marked in **Red** ***

Wed 30 December 2009: Specials version 4.2e-32

- Fix: Further fix related to UIDVALIDITY issues, that was crashing surgemail if you clicked on a messages that was the result of a cross folder search and this actioned the UIDVALIDITY reset of the surgeweb folder cache.

Tue 29 December 2009: Specials version 4.2e-31

- Fix: Drafts in the drafts folder that should be getting automatically deleted were no longer getting automatically deleted if you are not in the drafts folder. (ie usual case for autosaved drafts - broken with feature additions in 4.2e-26)
- Fix: UIDVALIDITY bug was actually a surgeweb fault mixing up folders on some commands. Mixing up of folders is fixed now, I'm pretty sure anyway... but will be doing further testing the next few days.

Sat 26 December 2009: Specials version 4.2e-30

- Fix: Implemented workaround in surgeweb UIDVALIDITY code, as bug in imap code (yet to be fixed) was resulting in surgeweb having to invalidate its indexes way to frequently (and a variety of other errors in interface as a result).

Wed 23 December 2009: Specials version 4.2e-29

- Fix: IMAP UIDVALIDITY based reset code in build 4.2e-26 would sometimes have multiple threads trying to reset a folder at the same time which would result in surgemail sporadically crashing in odd ways.
- Fix: The following character strings in a message ("&" "\$" "\$") would have led to odd things displayed in message display ("%the_msg%" displayed or duplication of the message header)

Sun 20 December 2009: Specials version 4.2e-28

- New: A new setting added "pref_nocontacts" to be manually added to a _user.dat file to troubleshoot a particular account for performance etc issues without any of the contacts information getting downloaded.
- Fix: Two signature handling fixes, using IE and popup message windows signature was not initialising correctly, and code to detect changes in signature text then switching signatures was not working correctly.
- Fix: If https is enforced and you connect using http, you are redirected immediately rather than redirected "after the http login page is mostly loaded".
- Fix: Mime constructs that did not have a space character between "Content-Type:" and the actual content type were not getting identified properly. The issue that highlighted this was email enabled printer/scanner whose attachments were not showing.

Fri 18 December 2009: Specials version 4.2e-26

- New: SurgeWeb now tracks IMAP UIDVALIDITY per folder (as it really should have done but never did), so "if UIDVALIDITY happens to change" on the server surgeweb will clear its message indexes and cached message files to avoid mixups in UIDS.
- New: A pref_reply email address gets used in the return-path header now too (it was already used in From header; and one still needs to explicitly manually specify the "allowed_from" in _user.dat to allow an account to use a from address that does not match email address)
- New: In the message list, messages in the Sent folder are displayed in grey when searching across folders to make conversations easier to follow.
- New: "Draft emails" usable as templates without them getting deleted on send. Surgeweb now saves draft messages with imap draft flag and opening any message with the draft flag opens it for editing rather than for display. Copying draft messages to any folder other than Drafts folder means they will not be deleted when sending. Most usable is to create a folder named "Templates" (which will update the menubar with an edit button like the Drafts folder) and use this to store "template draft messages".
- Fix: Clicking a folder surgeweb displaying new messages and then having background new_mail check happen later still displayed the audio / visual newmail indicator for messages that were already displayed due to manual folder refresh. This was confusing as you may well have already read these. The newmail audio / visual indicator only happens for new messages that arrive since last time the whole folder list was displayed.
- Fix: Deletion/move etc of a message then moving the cursor using the arrow keys in the direction that "the pending confirmation but hidden from display" message was no longer working (since the immediate hide in 4.2c-5). It was already "working again" as soon as confirmation was received the message was deleted / moved etc.

Sat 12 December 2009: Specials version 4.2e-24

- New: Surgeweb shared addressbook updates get mirrored, updates to contacts icon images (either shared or personal) now get mirrored too.
- Fix: Global and domain shared addressbooks were no longer editable with surgeweb - additional shared addressbooks defined using g_abook settings were not affected by this bug.
- Fix: If html tags were used in signature, this sometimes results in some parts of these getting displayed in a rather odd format at the start of your signature - particularly using IE but I think it may well affect other browsers too (introduced with signature features of 4.1d-5).
- Fix: Contacts odd situation fixes: Ampersand characters were no longer working in group names (result could not add contacts to groups with ampersand in name); Webmail addressbooks with brackets in name were not treated correctly; Webmail addressbook entries with invalid address entries (eg in email address field: "User Name") were not treated nicely; Single quotes were not being correctly encoded for webmail import (result was unusable contacts - if any customer contacts fail to load, check if they have a group like "John's group" and replace this with "John\'s group" in their user.abk file - if it still fails pass me the user.abk file for further analysis.)
- Fix: If quoted printable encoding was triggered (eg high ascii characters) during a message send, and the message had attachments and the message was not html the quoted printable header was added to the wrong message part resulting in display of "=20" at the end of message lines.
- Fix: Several surgeweb memory leaks fixed

Sat 5 December 2009: Release version 4.2d2-2

- Fix: Several fixes already in the 4.2e-* builds

Sat 5 December 2009: Specials version 4.2e-21

- Fix: Some combinations of contact address import were leaving the surgeweb contacts in an unusable state, addressbooks with this particular bad state will start working again with this build.
- Fix: (IE only) Made windows backspace key (same key as osx delete key) delete a message without browsing back to previous page in IE.

Fri 4 December 2009: Specials version 4.2e-20

- New: (non surgeweb) Changes in surgemail webserver implementation to improve performance on systems with large numbers of processors.

- New: Print button is available in customisation options for display on menubar in message list and in individual message display.
- Fix: (FF and Safari only) Bug in search field changes last week meant that in keyboard shortcuts (like N for new message etc) were getting actioned whilst typing in the search field.

Tue 1 December 2009: Specials version 4.2e-17

- New: Settings to globally disable newmail checks and inbox caching ('global_nocache true' and 'global_nocheck true' to be added in config_*.dat files) - useful for troubleshooting excessive load circumstances.
- New: Significant tidyup of the contacts import functionality as documented individually below.
- New: Improved & prettified import dialog with vcf, csv and webmail imports as part of same dialog. Contacts import is now a two step process, the first step uploads the file and shows how many new contacts and contact updates will be added, and then a second step to confirm or cancel the import.
- New: Webmail import can now be run multiple times without getting duplicates. Webmail distribution lists can be imported. Both webmail addressbooks and distribution lists get imported automatically and display of webmail contact tickboxes has been removed (old behaviour can be restored by adding "import_webmail manual" in config_*.dat files). In addition webmail imports can be triggered later using the import dialog.
- New: The most recent contacts import is added as a "Last Import" group. In addition to the "Last Import" group, the import dialog allows you to select another group to add your imported contacts to. Deleting a contact from a group now shows a dialog to choose between "removing contact from group" or "deleting contact" (ie removal from all groups). Deleting a group, now shows a dialog to choose from "delete group" or "delete group and all contacts in group".
- New: Additional minor tweaks and fixes surrounding contacts imports, including: contact groups now support single apostrophe character, crashing for non rfc compliant vcards.
- New: Allow the use of up and down arrow keys (to move the message cursor) directly whilst the search box has focus on the message list page without needing explicit message list selection. Pressing escape on message list level clears search just like pressing escape on search field.
- Fix: Logging out was no longer correctly terminating a session. The effect of this that was noticable, was that ticking the "remember me" on login page continued to work, even after the logout button was pressed.
- Fix: Firefox only error popup "unknown condition 2", if actioning a reply / forward to a message that was not yet cached browserside. The result was that the reply composition pane never fully initialised, and reply could not be edited / sent.
- Fix: Google chrome warning (see build 4.2e-14) was erroneously being displayed in IE when actioning a reply or a forward to a message that was not yet cached (ie using tickbox only to select inbox message with inbox caching disabled or other folder)
- Fix: Longstanding issue that in widescreen layout, using IE, and a scrollbar displayed on the message list the preview dropped downwards "offscreen".
- Fix: Reply-all was updating the message icon to a forwarded icon rather than a replied icon.

Wed 25 November 2009: Specials version 4.2e-14

- New: User.cgi "Change password" and "Filtering and spam control" options pages now available in the basic interface.
- New: Google chrome 4.0.233.16 workaround in build 4.2e-11 turned into a warning, as the workaround in build 4.2e-11 did not work to isolate the Chrome fault. I think it will work now but I have yet to encounter the issue since this change.
- Fix: Binary data in vCard imports was not getting correctly handled
- Fix: Handle leak in initial background download thread of large folders.
- Fix: Forward attached emails are now actually sent as MIME type "message/rfc822" rather than ordinary file attachments - this means that outlook natively opens the message. Most mail clients were already natively opening these based on file extension previously.
- Fix: Error handling in "initial folder download background thread" could crash surgemail if connection to imap server failed. Notably this could happen in a frontend - backend configuration if the backend system was shutdown / unavailable.
- Fix: Weird dialog interactions, notably when receiving a "session no longer valid dialog" when already in one of the more complicated dialogs, but also got hit in other circumstances. Was not clearing the dialog properly, resulting in very odd dialog contents in some cases.
- Fix: If webmail addressbooks had email addresses defined as psuedo distribution lists eg "User Name <email@addres1.dom,email@addres2.dom>" (this would have worked in some places of the the old webmail interface) and this was imported into surgeweb, this resulted in invalid entries in the surgeweb addressbook. The invalid entries could not be deleted or removed from the "Webmail import" group. These entries now get identified and corrected by turning them into surgeweb groups.

Sat 21 November 2009: Specials version 4.2e-12

- Fix: (non surgeweb) Minor tweak in [webdav implementation](#) (which btw is pretty much fully functional now) broke all surgemail web interfaces other than webdav

Fri 20 November 2009: Specials version 4.2e-11

- New: Delete added the the message context menu (right clicking message(s) in the message list)
- New: Workaround for Google chrome (notably version 4.0.223.16, but may affect other versions) getTime system function that is sometimes infinitely recursive, resulting surgeweb interface not actioning any ajax requests anymore, so seeming almost totally "locked up".
- New: Added [advanced customisation](#) documentation page for the "things you might want to change and are tempted to change in the tpl directory".
- Fix: Method for disabling Global & Domain shared contacts as per advanced customisation link above was resulting in errors for any actions rebuilding contacts information (eg import, message send, and a few others)
- Fix: Contacts fieldname clash was almost but not entirely resolved yesterday, resulting in errors when adding contacts to fixed groups and the contact getting added to ALL fixed groups when adding to custom group.
- Fix: Added the setting "labels_disable true" advanced customisation setting that can be manually added config_*.dat files to completely disable surgeweb labels in the interface.
- Fix: More visible and less grating cursor colors per default.
- Fix: Improvements to the cursor handling during contacts drag and drop - still further contacts drag and drop improvements planned soon.

Thu 19 November 2009: Specials version 4.2e-10

- Fix: Kinda nasty bug that resulted in unpredictable behaviour when trying to delete anything but the most trivial groups. Typical behaviour only some contacts removed from the group and the group not getting deleted, but could even result in surgemail crashing. This bug seems to have always existed.
- Fix: Contacts manager fieldname clash which meant that the fixed groups (by default Colleagues,Family,Friends) were sometimes not displayed and sometimes displayed with deletable icon next to them.
- Fix: If an account had custom deletable groups in personal addressbook, would fail to display contacts management interface straight after login - would start working later on though. (broken yesterday and now fixed)

Wed 18 November 2009: Specials version 4.2e-9

- Fix: There was no way of deleting groups in your personal addressbook (again since the changes in build 4.2e-4)
- Fix: Contact removal from a group was actioning a contact delete (since the changes in build 4.2e-4)
- Fix: Disabled status warnings when logging out to prevent background requests like inbox caching etc requests showing javascript warnings because they were terminated because you are logging out.
- Fix: (IE7 only) In Spam preferences selections were not getting "greyed out" in the second selection box as disabled options are not supported in IE7. It now appears like they are disabled by coloring individual options grey or black.
- Fix: Refreshing of webmail abk files for readonly display in surgeweb was not working properly with surgeweb's addressbook caching. Fixed.
- Fix: Contact picker was not working anymore on popup message windows.

Tue 17 November 2009: Specials version 4.2e-8

- Fix: Problem with the uploading of contact icon images fixed.
- Fix: Surgeweb's addressbook caching was not working properly with updates on the ldap and authentication database based addressbooks.
- Fix: Doing a search for a message across multiple folders, then selecting a message in the preview not in the folder you started in, then pressing "download all" attachments was resulting in bad request or empty file zip dependant on message uids.

Mon 16 November 2009: Specials version 4.2e-6

- Fix: Additional warnings and button hiding for webmail contacts and shared contacts in the html interface.
- Fix: Selection handling had been slightly broken.
- Fix: Safari only issue where on "some" of the surgeweb pages pressing enter on the relogin dialog ws not actioning the relogin dialog. (I believe it worked for all pages in most browsers and most pages in safari already)
- Fix: Edit draft was not working on the html basic interface.
- Fix: Using https from surgeweb to user.cgi (using the surgemail.ini setting "surgeweb_backend_web https://1.2.3.4:7443") was not working quite right, resulting in issues on the preferences page.
- Fix: Spellchecker fixes in 4.2e-4 had not actually been committed to source control yet. They are part of the build now.

Sun 15 November 2009: Specials version 4.2e-5

- Fix: Several bugs in the 4.2e-4 build fixed

Sun 15 November 2009: Specials version 4.2e-4

- New: Major rewrite of the surgeweb contacts handling to improve usability, improve efficiency and add a bunch of new features. Individual

changes and features documented below.

- New: Print contact vCard and print contact list features now functional.
- New: Basic interface now has a contacts management interface. Similar tree interface to the new Ajax interface contacts management interface. Only provides access to a limited set of contact fields (name / email / phone / address).
- New: Surgeweb serverside level addressbook caching enabled. This should significantly lighten the server loading (particularly on backend imap server) of surgeweb addressbook handling actions and should also make addressbook actions slightly speedier browserside. If stale addressbooks / strange behaviour is encountered this can be disabled by adding "abook_caching_disable true" in the custom/config*.dat files. Additional improvements still to be made here :-)
- New: Support for separate browse and search modes for addressbooks. The search mode is designed for addressbooks that are too large to sensibly store browserside. This feature is only available for the LDAP and authentication database based surgeweb addressbooks for now. For auth and ldap addressbooks defaults to auto, which means browse if <200 addresses and search if >200 addresses. Modes and limits configurable in the ldap and auth addressbook configuration options.
- New: LDAP based surgeweb addressbooks. Options much like authentication database based addressbooks. This is only available on windows builds currently but should be fairly easy to add to unix builds too. Let us know if you have need for this on unix.
- New: Authentication database based surgeweb addressbooks. Variety of configuration options available from the configure link on the abook page once an addressbook is saved as auth or ldap. One notable feature for auth based addressbooks is to limit the contents by access group useful to say have a "list of students" and a separate list of "staff" for example.
- New: Icon images can be added to any personal or surgeweb shared contact entries.
- New: Core of the rewrite made active. This splits personal and shared contacts, and splits "groups" into addressbooks and distribution lists, and presents this as a tree in the interface on the contacts page. In the background addressbooks are processed individually providing the facility for improved performance (performance improvements still to be enabled). The contact picker has been rehashed based on the same concepts and now has "All personal contacts", "groups from personal addressbook and directories" and "directories containing all shared addressbooks" the separate tabs. This change set has the potential to break any part of surgeweb but I have tested all things I can think of to work now... additional behaviour changes will be added soon.
- New: Multiple email addresses per contact are now all available / individually selectable in the contact picker and in the autocompletion dropdown list.
- New: vCard vcf contact files can now be imported using import in surgeweb pages or "tellmail import_vard ..." command. Both vcard and cvs import will now try and reuse existing contacts if it can (based on email address and username) so updates can more easily be applied without getting a zillion duplicate entries.
- New: Old webmail groups now displayed and usable as groups within surgeweb - still readonly, actual import of webmail groups still to be implemented (soon)
- New: Queued web request channels logs are available under the green info icon to troubleshoot permanently rotating icon issues.
- Fix: Minor tweak to the handling of numerical bullet points in html messages (as generated by MSWord) to display them tidier.
- Fix: Two recipient handling limits fixed in surgeweb. 1) During send if any of the to/cc/bcc fields were bigger than 1kb (20-40 addresses) the whole field was getting dropped; 2) display of recipients was limited to 5kb (100-200 addresses) and this was the maximum number of recipients replyall would put in the addressing fields. Now should be unlimited within surgeweb. The following have been tested to work fine up to 1000 recipient addresses per email (at which point surgemail's g_rcpt_max kicks in): recipients in to/cc/from fields, sending to groups (either browserside or serverside expanded), display of recipients list within email, recipients displayed when pressing reply all. note: one fix is template file based, the other surgemail binary based, so a full distribution is required to apply fix.
- Fix: Spellchecker fixes: some suggestion words were resulting in javascript error, and now deals better with apostrophes in words eg "Couldn't"

Thu 5 November 2009: Beta version 4.2d-1

- New: Ability to customise mobile interface logo image (manually add "mobile_custom_banner true" and then place banner_mobile.jpg in custom directories)
- Fix: Made sureweb more tolerant of spam body modification, spam body modification resulted in surgeweb pretty much always truncating the message for display.
- Fix: (non surgeweb) Several non surgeweb fixes as per [surgemail changes](#)

Wed 28 October 2009: Specials version 4.2c-5

- New: Messages get hidden immediately on message delete / move to make interface seem more responsive (tweaked version of setting introduced in build 4.2b-11). Can be disabled in advanced preferences.
- Fix: Minor rewording on "Filtering and Spam control" preferences page
- Fix: Made "lock_name_auth" apply to basic interface too.
- Fix: Actually made the delete key get actioned at the global level. After recent keyhandling changes just the delete key still needed explicit message list selection.

- Fix: After relogin using "relogin dialog" due to session timeout, save draft showed error even though the save actually succeeded. (may also have affected other commands)

Sat 24 October 2009: Specials version 4.2c-4

- Fix: Pressing NotSpam on messages in Spam folder was not training & cleaning messages correctly, sometimes resulted in multiple copies of the message in inbox, sometimes even the wrong message.
- Fix: Pressing Spam / NotSpam / Allow Once / Block Forever results in javascript error. Note the action was still actioned and a folder refresh gets display synchronised again. Not sure when broken, but is broken in 4.2a3-3 production release.

Sat 24 October 2009: Specials version 4.2c-3

- New: Tidied various aspects of the Preferences tabs, including list of keyboard and mouse shortcuts and removal of not yet implemented items from options menu.
- New: Surgemail disables relevant parts of interface for (g_)user_access settings "pass,fwd,fwdonly,exceptions". Also if none of the Filtering/Spam control features are enabled the tab gets completely hidden.
- New: New surgeweb setting "lock_options_basic true" (manually add to custom/config_*.dat) - Hide the screen layout and advanced tabs in Preferences to prevent users from "playing" with them.
- New: New surgeweb setting "lock_name_auth true" (manually add to custom/config_*.dat) - Prevent end users from changing their "from address details" and update full_name from authentication database on every session login.
- Fix: Emails with from addresses with a single quote character in the actual address were failing to display, and failing during "conversation history" searches. Both fixed.

Fri 23 October 2009: Specials version 4.2c-2

- New: Settings to refresh and target right column based google ads. [more info](#)
- New: Google Chrome 3 browser bugs warning, with optional drag and drop disable. [more info](#)
- New: New surgeweb setting "smtp_ssl_required" settable on the customisation page to force SSL on sending SMTP connections without the need to explicitly specify IP:port for each domain using the surgemail domain setting surgeweb_backend_smtp.
- Fix: Filtered out Cmd-Q so it has standard behaviour on OSX.
- Fix: Context menu on folders through "Ctrl clicking" was still not working on OSX.

Fri 16 October 2009: Specials version 4.2b-12

- New: Sending as text / html can be clicked to toggle mode during compose (interface probably still needs improving though). Several other related fixes including setting of html mode if color changes are made. Not setting of html mode if "remove all formatting" is clicked. Improved detection of replying in html mode if message is multipart alternative.
- Fix: The csv contacts import was no longer working due to recent cross site scripting changes. Fixed and now supports a lot more outlook fields to import and is smarter about dealing with fields it does not know about if "Add unrecognised data" is ticked.

Thu 15 October 2009: Specials version 4.2b-11

- New: Rewrite of selection handling (mainly keyboard selections), fixing many things - including: top & bottom end of selection handling better, bunch of message actions (like keyboard based select all / delete / cache etc) can be done without first needing explicit message list selection, Ctl-A can be used to select all message body text in preview pane (click preview first to set focus), support for OSX Cmd key and Option modifier keys instead of Ctl to match normal OSX key behaviour, support for Ctl click on OSX to get context menus, Windows "menu alt keys" can be pressed without message actions triggering.
- New: Sort order remembered per session serverside, so new message notifications do not reset the client side displayed message sort order to reverse date. Also option to make sort order "sticky" so it will remember what it was set to (default is always revert to reverse date on a new login). Also sorting by to / from addresses fixed so it removes non printable characters like quotation marks before sorting.
- New: Experimental option to hide messages immediately instead of greying them out first when deleting etc (setting needs manual addition until we decide whether this is actually a sensible feature;-)
- Fix: Crash during html washing if a particular odd html construct was encountered
- Fix: Forwarding a message with attachments or forward attach was losing the attachment at send time if using the basic html interface.
- Fix: Tweaked the "Single line on reply" header slightly

Fri 9 October 2009: Beta version 4.2a3-3

(patched build for production release with several fixes already in 4.2b-* builds)

- Fix: Doing a "download all" of multiple attachments with no file extension and the same name crashed surgemail.
- Fix: Forwarding a message with attachments or forward attach was losing the attachment at send time if using the basic html interface.
- Fix: (non surgeweb) Friends processing bugfix

Tue 6 October 2009: Specials version 4.2b-7

- New: Header to quoted message text on replies is now a single line - there is an option to disable this and display same multiline header as forward like it used to.
- New: Made cross site scripting vulnerability checking more rigorous. Note: be on lookout for these that may have been introduced as a result: 1) interface actions that not longer work 2) html messages that do not show correctly (due to stricter HTML washing, please pass the emails in original format from disk to surgemail-support@netwinsite.com for analysis)
- New: OSX Fluid user script to have multiple concurrent surgeweb accounts as "permanently online web applications" with always up to date unread count in dock, automatic relogin on session timeout, and automatic start on bootup (I'll write some instructions and add to help and post a link here)
- New: Native support for "unread message count" (just like Apple mail - in OSX dock and application switcher) when running under OSX Fluid Site Specific Browser. (note: Fluid is a useful way to make any web page behave "like an application" under OSX)
- New: "Nice icon" on iPhone when creating a "bookmark on home screen" to any of the surgemail web interfaces
- New: Support for Thunderbird fields for the contacts import of csv files
- New: A surgeweb reply to a dbabble generated email message is now sent plain text rather than multipart alternative html.
- Fix: Doing a "download all" of multiple attachments with no file extension and the same name crashed surgemail.
- Fix: Issue with the saving of preferences that enabled these advanced settings (sticky_menus, no_fancy_rcpt, no_replycolor, no_filtermenus) "for the rest of the session" if any preferences save was done.
- Fix: Make the message list icons (attachment, priority, security) & label alignment work nicely in IE7 (& IE8 compatibility mode)
- Fix: Make the delete key on the OSX apple keyboard delete a message and not "browse back" out of surgeweb.
- Fix: Rewrite of the menu handling code fixing a bunch of menu problems. Most notably right click context menus not working at all on Firefox and Safari. But other issues too. Menus should now mostly all work either as a "click(hold) - drag(to menu selection) - mouse up" or "click(with mouseup) - drag(to menu selection) - second click" action on all browsers.

Thu 24 September 2009: Beta version 4.2a2-2

- Fix: (non surgeweb) Recently added crashing bug in the archiving code (only seemed to affect some systems).

Wed 23 September 2009: Beta version 4.2a-1

- Change: References to "Ajax interface" changed to "Standard Interface" on login page and interface switching buttons. Documentation not yet reviewed.
- Fix: Clicking contacts search field clear it ready for input (was broken some time back)
- Fix: No content was getting displayed for html messages with a <head> but no </head> tag (technically invalid html)
- Fix: Filehandle leak during send of plain format=flowed messages (introduced with format=flowed plain text features of build 4.1b-14)

Mon 21 September 2009: Specials version 4.1d-6

- New: Added example to extend.js of making the change password button point to your own password administration web page
- Fix: Problem with the language translation file. Any "brand new" surgemail installs since 4.1c-1 will have a badly formatted "surgemail/lang_web.dat" file (upgrade installs are not affected). Symptoms are garbage entry in "language drop down" on front page and "Remember me" displayed to right of language dropdown. For now fix is to manually replace the surgemail/lang_web.dat with the version from any new distribution (4.1d-6+) as this file does not get replaced on upgrades currently.
- Fix: Browser check tweaked to make it better able to detect Safari based browsers (such as Fluid)

Sat 19 September 2009: Specials version 4.1d-5

- New: Able to easily add signature to "just a few" or "not a few" messages. Selection done on compose pane and if "remember" sticky options are used on the compose pane this is remembered for next message. Without "remember", the show / noshow default can be set in Options - Preferences.
- Change: When replying to plain text message, plain text reply (format=flowed mode) is only enabled if message actually arrives with format=flowed formatting, else html (multipart alternative) is used to keep dumb Microsoft mail clients happier.
- Fix: Recently added "Remember" sticky compose options work on popup message windows too now
- Fix: Minor changes to the way the html editor is initialised. Timing bug, which sometimes resulted in Google Chrome sometimes having an uneditable compose pane should be fixed. If anyone encounters problems with the editor (particularly in older versions of Firefox) please let me know.
- Fix: When using "download all" attachments feature, handle it more nicely if the all the attachments actually have the same filename

Tue 15 September 2009: Specials version 4.1d-4

- Fix: Have surgeweb "reply indenting" work better with plaintext messages with spaces between quote characters as per "> > Original". Notably existing netwin webmail formats it replies like this.
- Fix: When sending emails with an ampersand character '&' this would get replaced by a random letter in the text part of sent messages. Bug has always existed but is more noticeable now that surgeweb preferentially sends in plain text. Html parts of multipart/alternative messages not affected.

Mon 14 September 2009: Specials version 4.1d-3

- Note: Surgeweb will clear and regenerate its imap folder indexes first time a folder is used after this build has been installed (this may take up to a minute or so per 10000 messages in a folder but is done in the background)
- New: More options for hiding some of the interface elements (particularly surgemail spam related warnings if not using surgemail spam handling) using extension css file: extend.css
- New: Separate Reply indicator into Reply / Forward / Reply+Forward
- New: Attachment, priority, security indicators as part of message list
- New: SurgeVault encryption integration for message sending (PRERELEASE - contact surgemail-support@netwinsite.com if you wish to try this)
- New: "Remember" tickbox in the message compose options that allows the message compose options to be "sticky". ie. display of cc/bcc/options and state of priority/confirmation
- New: Automatic timezone detection from browser
- Fix: If messages in the message list were clicked before list was fully displayed this resulted in javascript error.
- Fix: Tidied "marking as read" handling. In particular, if no preview was present or web mode was used it would mark as read when it shouldn't, or was not marking as read when it should in some situations.

Mon 7 September 2009: Beta version 4.1c-1

- Fix: Text part of multipart/alternative was missing a cr/lf if the event it did not need to be quoted printable encoded
- Fix: Surgeweb generated plain/text messages with attachments had an extra mime header which displayed as part of the text content

Fri 4 September 2009: Specials version 4.1b-16

- Change: Keyboard "message send" shortcut is now Ctrl-Shift-Enter, it seems Shift-Enter was getting frequently accidentally hit just like Ctrl-Enter. Please let me know if it is improved now.
- Fix: Surgeweb "eating whitespaces sometimes". Primary problem affected sending using firefox after doing a copy and paste of text - affected both text and html parts of multipart mime messages. In the editor the message looked correct at time of sending. Other browsers not affected. Secondary problem affected all browsers but only possible sometimes if sending a plain text part generated from html display in editor (thus only possible since addition of plain text sending in 4.1b-14)
- Fix: Addition of square brackets around an email address in surgeweb prevented email from getting sent and resulted in contacts being completely broken. Note: on any accounts that contacts is broken for this reason, contacts should start working again with this fix.
- Fix: Html messages with multiple style blocks were no longer getting handled correctly (as of build 4.1b-11)

Mon 31 August 2009: Specials version 4.1b-15

- New: Create link button in the html editor toolbar
- New: Surgeweb html editor now supports setting of font color and background color in the message editor (includes blogs editor)

Sat 29 August 2009: Specials version 4.1b-14

- Note: If message formatting oddities are encountered with these changes please send me the actual message with bad formatting, and very importantly also the original message that was replied to / forwarded. I need both in raw format from disk for the troubleshooting of any issues here.
- New: Parts of the documentation updated. Notably these pages added: [contacts documentation](#) (includes shared contacts), [faq page](#) and multiple server clustering [configuration notes](#). Also installed surgemail help fixed wrt new netwinsite.com layout changes.
- New: Support for format=flowed plain/text only messages. The ajax interface surgeweb will default to sending messages in plain/text (rather than multipart related with html). Editing is still gets done in the html editor with colored reply indenting etc. If you use any of the html markup on the toolbar the message will automatically switch to getting sent as a multipart/related message with both text and html parts. If you reply to a message that starts of as an html message your message will be sent as html. (possibly need some more finer grained control but I suspect this is all that is needed)
- New: Surgeweb outputs the textpart of as nicely formatted format=flowed plain/text. In addition to this, if the text part contains a few UTF-8 characters quoted-printable content transfer encoding is also used. If the text part contains lots of UTF8, the whole text part is base64 encoded instead.
- Fix: A little of this code was mistakenly active in Thursdays builds which would have resulted in oddly formatted plain text parts in multipart

alternative html messages (ie mostly would have been invisible to users).

Thu 27 August 2009: Specials version 4.1b-12

- New: CSV import can be imported direct to any shared addressbook you have permission to access. Number of records imported gets displayed to user.
- Change: Send message keyboard shortcut changed from Ctrl-Enter to Shift-Enter, it just happened too often that you pasted with Ctrl-V and followed that with the enter key and having surgeweb recognise that as "send the message now" because your finger was not off the Ctrl key yet.
- Fix: Minor fixes to display of multilevel quoted replies.
- Fix: The csv import was not importing the last field specified on the formatting line of the CSV file. Also if no ',' are found it will try to use the ';' as a field separator instead.

Thu 27 August 2009: Specials version 4.1b-11

- Fix: Crashing bug during friends bounce reply processing if the email had certain formatting. (introduced 4.1b-6)
- Fix: Make sure the surgeplus / user.cgi autologin works on frontend / back server combinations
- Fix: Text part processing issue that had some odd consequences as a result of browser quirks. If the source of a reply was a text part of another message (as opposed to html part), it would remove spaces between some words "during the send" even though it looked correct at in the browser at time of sending (only affected some browsers). The same issue was responsible for the addition of spaces after certain text lines which meant that the '\ ' line continuation character could not be effectively used in messages.
- Fix: Html part of multipart messages sent by surgeweb are a little tidier

Mon 24 August 2009: Specials version 4.1b-10

- New: Ability to hide "Remember me" tickbox on the front page - again requires css customisation and sample code in extend.css.
- New: Ability to display Filestore button as well as / instead of the Photos button to get to the top level of the surgeplus filesharing (needs to be enabled in extension css)
- New: Enabled skin styling and extension css / js for the basic interface and the stylings defined for ajax should apply mostly here.
- New: Received multilevel format=flowed text only messages get displayed nicely using indentation mechanism below.
- New: Blockquote based indentation when replying to make levels of reply easy to spot, with automatic "breakout" to outer level on pressing enter with cursor in text. As displayed in surgeweb first reply level is always blue, second reply level green, then light brown and the rest light grey (text color explicitly set in html does override this though). This mechanism is send/receive compatible with the reply indentations as displayed by other clients (notably thunderbird, osx mail, gmail). Note the colouring does not get explicitly set so what the destination message "looks like" in terms of colors depends on the destination mail client. (Only outlook does not have much of a visual cue yet...) There is a setting to revert old behaviour in preferences.
- New: (non surgeweb) html spam status email has facility for normally disabled "report all as spam" action link.
- New: (non surgeweb) html spam status email also displays summarised To header information to detect messages to multiple recipients / undisclosed recipients / bcc's etc.
- Fix: Crashing bug in the body searching code if the tcp connection was prematurely broken.
- Fix: If using g_maildir_netwin, pressing "Send >>Done" on a message already in the Done folder resulted in an error (which meant it was easy to resend the message as it was already successfully sent).

Fri 14 August 2009: Specials version 4.1b-5

- New: Chrome spell check fix earlier today, broke the chrome focus to message body on reply/forward etc code - focus now fixed.
- Fix: Folder names now have full UTF8 support. This means any folders with non ascii characters created in other mail client should show in surgeweb and visa versa. Folders with utf8 characters previously created from surgeweb were stored in imap with invalid foldernames, these are corrected and ambiguous characters replaced by '?' and suffixed by the text "- invalid name N".

Thu 10 August 2009: Specials version 4.1b-4

- New: Google chrome native browser spellchecker now works in the message editor when composing messages
- New: ICONV international / UTF-8 character set handling for linux64, freebsd7, solaris8, solaris_x86
- Fix: In basic html mode, delete & move when a message is displayed have never worked and move copy message in message list were recently broken.
- Fix: Menus that "would not go away" broken several weeks back. Let me know if there are any menus that disappear too soon now...
- Fix: Automatic contact addition when replying to emails with from header spread over multiple lines (notably happened for international characters =?iso encoded international characters from hotmail) would result in broken addressbooks. Fixed for new address addition, and surgeweb addressbooks previously broken like this will start working again too.
- Fix: Strip inline html tag "position:" styles. Notably makes some yahoo groups messages display nicely in IE7 (message text was bleeding all

over the whole page and not scrolling).

Thu 23 July 2009: Beta version 4.1a-1

- Fix: Messages containing unicode line separator characters would not display in FF & Safari
- Fix: Handle leak when attempting to send a message without any valid recipients having been specified
- Fix: Explicitly specifying the domain to login to using domain_ex variable made a bit nicer
- Fix: Setting to disable fancy recipient addressing did not work

Thu 16 July 2009: Specials version 4.0x-4

- New: IMAP based full body search is implemented

Tue 14 July 2009: Specials version 4.0x-3

- New: SurgeWeb disables appropriate parts of the surgeweb interface in response to 'friends' and 'spam' user_access settings.
- New: Changed the quota display in the Mail panel header to include quota limit.
- New: Make html editor automatically switch from markup to normal mode before sending/spellcheck/saving emails; in signature editor; and editing blog posts.
- Fix: Made edit in markup mode work for multiple concurrent messages in tabs.
- Fix: Messages with attachments or spam control buttons would not display in folders with a single apostrophe in the name
- Fix: Text "blurriness" removed from selected messages text in message list in non high contrast cursor mode
- Fix: Recent bug that had surgeweb display "(none)" in from column if a surgeweb label was applied (imap label was fine) or a message had a X-SpamDetect header (big-endian systems only ie Solaris sparc and OSX PPC)
- Fix: Links in top right of messages (prev/next, conversation, popup, raw/text/html) did not work if searching across multiple folders and dealing with a message not in the primary folder you started your search from
- Fix: IE only focus issue where edit fields in interface would sometimes become unselectable, very visibly the search box - and also other things like text could not be selected for copy paste. Happened under some conditions when caret focus was within an iframe that was closed (message body edit, signature edit, or any field in user.cgi dialog). Previously doing an action that explicitly set focus would already reset this focus issue (eg. compose, reply, forward, labels menu etc)
- Fix: Spellcheck "word correction menu" menu had border removed but shadow effect was not yet getting added.
- Fix: Reply actions (mark replied, moved to done folder etc) to messages listed in a folder based search (searches with folder column displayed) were getting applied to the wrong message. (first message in folder rather than the correct message)
- Fix: Labels menu was displaying "undefined"
- Fix: Labels menu was broken until you logged in again if you opened the labels menu on an account with no labels, closed the labels menu without creating a label and tried to open the labels menu again.
- Fix: Using 'ajax panel look' with borders, a gap appeared between top and left edges of dialog boxes and the shadow effect.
- Fix: Crash if another IMAP client had set custom IMAP labels before surgeweb had ever seen a folder and there were more than 200 messages in the folder in question.

Mon 6 July 2009: Beta version 4.0w-1

- New: Quota information gets displayed in basic interface too
- New: always_show_quota setting to allow quota to be displayed in interface if <80% of quota is being used
- New: Ability to create, delete and rename folders from the right click context menu on folders in left panel
- New: Changes to surgeweb's "spam" / "notspam" reporting. All [combinations listed](#) now do spam training. Spam training now works without the use of "g_imap_friends". (g_imap_friends is still required to view and deal with surgeweb's "Spam" folder)
- New: Menus prettified with shadows
- New: Message "copy/move to folder" menu is "searchable" for a specific folder now & keyboard navigable too (just like the Labels menu already was)
- New: (non surgeweb) status_html.eml status message is more complete (logging more complete, and now has "Block" link - report as spam and add address to usercgi block list)
- New: Closer integration between "quick search" and "folder search", [see help](#) . Menu selecting them is remembered as a preferences and defaults to Quicksearch again.
- Fix: Basic & Mobile interface "sent" folders were listing from address rather than to address
- Fix: "Challenge unknown senders" configure link had been broken some weeks back, although was going unnoticed in some browsers
- Fix: Tidied some of the warnings and error messages

Tue 23 June 2009: Specials version 4.0v-22

- Fix: "Tamed" the "folder count updated" message added yesterday that was getting displayed more often than intended

- Fix: Crash when doing notspam action in inbox (introduced a few days back)

Mon 22 June 2009: Specials version 4.0v-21

- New: Ability to set message priority when editing messages for sending (toggle options on compose page first)
- New: Ability to set read confirmation / delivery confirmation requests when editing messages for sending
- New: Unread message count display gets refreshed for all folders at login and every 10 minutes during a session
- New: Folder context menu that allows manual folder refresh of one or all folders (downloading headers for searching and flags if they have been set with another mail client)
- New: "Extra imap refreshes" setting works and does full flags update periodically (folder click requiring full update on large folders may be noticeably slower particularly on busy server)
- New: Slightly changed the way the flags updates get done making it faster 99% of the time
- Fix: To the way surgemail does its new mail check which sometimes meant that clicking folders shortly after login would not show new messages for up to 15 seconds sometimes
- Fix: Arrow key based message selection got reset after clicking message when message list spanning multiple folders was displayed
- Fix: Made background gradient prettier
- Fix: To the use of labelling on non surgemail IMAP servers

Sat 20 June 2009: Specials version 4.0v-20

- New: "Send >>Done" is now actioned as a single html request rather than two separate html requests.
- New: More sensible handling of the default flags created by other IMAP mail client (spam flags get hidden and recognised flags get initialised with label text and colours) and more functional "mail client mapping" on flags management page.
- Fix: Labels search in menu only applies to start of label string
- Fix: Occasional redbox "unknown javascript errors" when sending from popup windows after pressing the reply button within the popup window (fixed I think, as I was not able to repeat the fault myself)
- Fix: Label creation through menu "lower cased" the whole labelname (fix part two)

Fri 19 June 2009: Specials version 4.0v-19

- Fix: Label creation through menu "lower cased" the whole labelname
- Fix: Selection handling in message list when creating labels was confusing
- Fix: Message move to folder button / more actions menu was no longer working (broken with implementation of labels menu)
- Fix: Friend pending silent mode messages were not getting purged at g_friends_pending_keep days
- Fix: Recipient context menu / clicking recipient address for new email messed up ones own from address requiring relogin to surgeweb to correct it
- Fix: Spurious warning displayed when actioning contacts "more actions" menu items from Chrome or Safari
- Fix: Labels and Searches trees remembers collapsed/opened state (just like the "More Folders" already does)
- Fix: Labels color picker tidied

Wed 17 June 2009: Specials version 4.0v-18

- Fix: Messages copied to Sent folder when sending have unread status. (broken with labels implementation in 4.0v-17)

Wed 17 June 2009: Specials version 4.0v-17

- New: Full surgeweb message labelling (compatible with other IMAP clients) implemented. Allows multiple labels per message stored as imap flags or message headers. For more info see [online help](#)
- Fix: About four fixes to existing functionality not related to message labelling that I came across during development. Primarily related to searching and character encoding.

Wed 10 June 2009: Specials version 4.0v-14

- Fix: Much improved the dropdown autocompletion speed for popup message windows under IE
- Fix: Fixed a deadlock situation where a particular failed IMAP command would prevent further surgeweb index file updates for that folder until surgemail was restarted
- Fix: Reply button on popup message windows was no longer working (broken 4.0v-11)
- Fix: First attachment in popup message windows would fail to attach (broken 4.0v-10, when quota warning was added)

Mon 8 June 2009: Specials version 4.0v-12

- Fix: Further search folders fix

Sat 6 June 2009: Specials version 4.0v-11

- Fix: Purge of an empty Spam folder no longer results in error
- Fix: Search folders Single / Recent / All button was broken in 4.0v-10

Sat 6 June 2009: Specials version 4.0v-10

- New: Quota information is displayed if green status icon is clicked. If >80% quota is used an additional warning is constantly displayed beside the status icon.
- New: (non surgeweb) Improvements to html spam report email
- Fix: Re / Fwd stripped from subject searches to also match the original message in conversations

Wed 3 June 2009: Specials version 4.0v-8

- New: Search now is able to search across multiple folders, actions (delete etc) can be applied to messages listed in a search that span multiple folders. Note as it stands search does not update indexes - so messages added to folders since folder was last opened in surgeweb will not be found yet
- New: Search defaults to folder search (option of: current folder, recently accessed folders; or all folders). Quicksearch "browser side search" is still available from dropdown or pressing the "q" key
- New: Serverside searches can be saved (aka smart folders) for future one click access
- New: Messages have button to search for messages in multiple folders by subject (aka "thread") or history by email address (ie full "conversation history" with a user)
- New: Right click context menus on recipient addresses, message list, message stars
- Fix: Several minor surgeweb fixes and several non surgeweb fixes

Fri 29 May 2009 : Specials version 4.0v-3

- New: Ability to flag messages
- New: Change password button on the top level preferences page
- New: Last account activity time and ip address on the preferences page (to allow users to detect unauthorised surgeweb use)

Thu 28 May 2009 : Patched Release version 4.0u3-3

- Fix: Warning if forwarding messages with multipart/related images that images will not be sent as part of the message (not implemented yet - should do redirect or forward attach for now)
- Fix: Several more small non surgeweb fixes

Wed 27 May 2009 : Version 4.0v-2 & Patched Release version 4.0u2-2

- Fix: Html interface compose receives signature too
- Fix: Html interface refresh and purge links work
- Fix: Html interface reply / forward etc shows from address fully
- Fix: Two processing fixes for plain text message text with paragraphs larger than 1KB - would truncate & mess up text part paragraph text and may also have been able to cause a crash
- Fix: CSS processing fix - if the processing buffer boundary fell on certain css tokens surgeweb would completely fail to display the message
- Fix: non surgeweb fix that prevented mail getting sent to domains that contained just the letters a-f

Wed 20 May 2009 : Beta build version 4.0u-1

- New: Surgeweb.log accessible on the surgemail admin interface log search page
- New: Reply and forward using basic interface add an '-- original message --' header in the body prior to old message
- Fix: Changing text message urls into links made a bit smarter
- Fix: Made display of html interface page "less flickery"
- Fix: Login page language translation warning removed when selecting languages (although most of the internal phrases of surgeweb are not part of the translations yet)
- Fix: Html interface autologin to surgeplus / user.cgi works again (broken with backend autologin features some weeks back)
- Fix: Address autocompletion works again in the basic interface (broken recently with the additional sorting to most mailed order)
- Fix: Html edit window when editing a message now resizes to fill the browser window
- Fix: Signatures were not getting added to emails when sending with basic interface
- Fix: Signatures get limited (and user warned) to 1KB - limit always existed but messed up configuration files if signatures larger than approx 970 characters (including html tags) were saved. Larger signatures will be supported soon but requires further development.
- Fix: Message popin / popout no longer adds additional copies of the signature

Thu 14 May 2009 : Beta build version 4.0t-4

- New: Popup warning if logging in to ajax with unsupported browser; and admin configurable action: unsupported_browser [allow | deny | basic | webmail]. Where basic and webmail login to other interface instead.
- Fix: Encoded attachment filenames should now be getting correctly handled
- Fix: Sending messages with basic HTML interface if you just had a "Name" specified in "options - Reply name / address" was not working (blank or fully specified name and address was already working).

Tue 12 May 2009 : Beta build version 4.0s-1

- Fix: Several further fixes to contacts handling

Mon 11 May 2009 : Beta build version 4.0r-7

- New: Autocompletion dropdown list is sorted with most frequently mailed at the top
- New: Surgeweb shared addressbook admin interface configuration page (search for 'abook' in config settings) now has instructions
- New: Webmail contacts import - webmail contacts can be individually imported, or all webmail contacts can be imported in one go
- New: Contacts - Shared contact group membership can be edited through surgeweb interface - by editing group edit field or by drag and drop to existing groups defined in shared contacts
- New: Contacts - Top N favorites increased from 10 to 20 and is customisable if desired
- New: Contacts - Instant Messenger field is actually displayed and works
- Fix: Contacts - Birthday field now works correctly
- Fix: Contacts - Contact auto addition clientside based on first ever send to an email address does not get name and email address reversed for duration of surgeweb session
- Fix: Contacts - Pressing cancel button when editing refreshes browser side information with unchanged contact information
- Fix: Contacts - Deletion of a single item of multiple record field works as it should
- Fix: Contacts - Confusion between display and editing of personal and shared addressbooks for contacts with same id (note: "id" generally but not necessarily the same as the email address)
- Fix: Contacts - Favorites now actually has most mailed, and recently mailed contacts in it (it was getting incorrectly sorted - big oops)
- Fix: Contacts - Manually adding a contact to favorites resulted in it being displayed twice in the list of favorites
- Fix: Contacts - Organising contacts in groups would lose detailed information for the contacts in question
- Fix: Certain group deletion actions could result in crash
- Fix: Certain MIME construct could result in crash

Tue 5 May 2009 : Beta build version 4.0q-1

- Fix: Some character set headers were not correctly decoded and as a result some characters were not recognised when converted to UTF-8
- Fix: Further fix to outlook csv import functionality

Mon 4 May 2009 : Beta build version 4.0p-14

- Fix: minor fixes / changes to html spam status email
- Fix: multipart/mixed messages with no text part and unnamed image get displayed correctly

Sat 2 May 2009 : Beta build version 4.0p-13

- New: (indirectly surgeweb) support for fancy html user.cgi spam status email with direct clickable links to release messages, delete messages and purge the spam folder (requires updated status.eml not yet in distribution)
- New: (indirectly surgeweb) user.cgi status email does no longer deletes user mail processing log file entries when spam & log file status message gets sent
- Fix: Display of non RFC compliant text emails could result in surgemail crash
- Fix: Outlook CVS import was not handling the last field (on each line of the import) correctly
- Fix: Spell checking certain words could crash surgemail
- Fix: Minor fixes to the surgeweb info in the tellmail status output

Fri 1 May 2009 : Beta build version 4.0p-12

- Fix: In the message editor, starting a line with a tab character turned the paragraph into preformatted. As a result disabling the wrapping.
- Fix: "Sessions that timeout serverside" fix was broken and as a result logged out all sessions once an hour

Tue 29 March 2009 : Beta build version 4.0p-10

- New: tellmail status now has surgeweb session statistics
- Fix: Sessions that timeout serverside get tidied up nicely
- Fix: Several memory leaks in surgeweb code
- Fix: IMAP filehandle leak in some IMAP features that surgeweb uses more than any other mail client.

Sun 26 April 2009 : Beta build version 4.0p-6

- Fix: Another surgeweb filehandle leak resolved

Sat 25 April 2009 : Beta build version 4.0p-5

- Fix: Made webmail contacts as stored by old webmail smooth template display nicely in surgeweb
- Fix: Crash if webmail addressbook files were corrupt in a particular way
- Fix: Having certain characters in subjects of "Forward Attach" was failing to actually attach the messages to be forwarded
- Fix: Optional custom menubar button "Forward attach" was not doing anything
- Fix: Links on mobile interface to switch to Basic and Ajax interface work again
- Fix: Custom configuration in "Preferences - Filtering & Spam control" is blue not red
- Fix: Several file handle leaks (certain conditions in: in forward attaching, and spam training multiple messages at once, saving of drafts)

Tue 21 April 2009 : Beta build version 4.0o-1

- New: Able to find surgeweb customisation page settings using the "Find config setting" in admin interface

Mon 20 April 2009 : Beta build version 4.0n-6

- Fix: Several memory leaks fixed
- Fix: File handle leaks if: 1) message failed to send; 2) replying to message in text mode (ie using mobile interface)
- Fix: Multibyte UTF-8 characters inline in an email no longer trips up the spell checker word synchronisation
- Fix: Display of messages in "web page" mode works again

Sat 18 April 2009 : Beta build version 4.0n-2

- Fix: Files sent as emails (by office 11) by right clicking from the shell / browser windows and were not getting correctly displayed by surgeweb as they have odd MIME constructs.
- Fix: Customised "Not Spam" button on menubar was getting actioned as "Spam" - button on message and More Actions menu was fine.
- Fix: Crash when logging in to large folder (>200 messages) for first time ever using surgeweb (introduced yesterday as part of "Friends Pending" customise feature)

Fri 17 April 2009 : Beta build version 4.0m-1

- New: Started on interface multilanguage support - main login page texts go through the language lookup. (surgeweb strings use surgemail\lang_web.dat and surgemail\lang_bin.dat)
- New: Ability to customise the imap name of "Friends Pending" (surgeweb "Spam") folder - must modify BOTH g_friends_pending_name and surgeweb imap_spam_folder
- Fix: Messages with many hundreds of recipients were not getting displayed at all. Now display message with truncated recipient list (still requires better fix)
- Fix: Encoded headers (most notably subject header) sporadically crashed surgemail (only affected solaris8, linux64 and freebsd7 builds)
- Fix: UTF-8 (and & character) support in webmail addressbook nickname field for the read only display of webmail addresses in surgeweb
- Fix: UTF-8 support for csv contact import - notably this also means that First Name gets correctly imported when importing csv file generated by OE / Windows Live Mail.
- Fix: Logging in with a space character prefixing a valid username crashed surgemail

Tue 14 April 2009 : Beta build version 4.0l-3

- Fix: Surgemail crash on every surgeweb login under Solaris 8 on some servers (could possibly have affected other platforms)
- Fix: Surgemail crash on surgeweb customisation page save linux 64 (could possibly have affected other platforms)
- Fix: Surgeweb customisation page "non default vdomain warning" no longer displayed on first display of customisation page
- Fix: Safari 4 and IE 8 now pass the login page browser check
- Fix: Drag and drop of messages and contacts works in IE 8
- Fix: Some spaces were getting "eaten" from some message text (certain html constructs generated by MS exchange) noticeable during display of and sending of messages
- Fix: Drag no longer starts from message list tickboxes (avoids accidental loss of multiple selection)

Thu 10 April 2009 : Beta build version 4.0k-1

- New: Per user ability to disable whether contacts are autoadded and tracked for favorites when sending email (disabling this is NOT RECOMMENDED)
- New: Per user customisation of date and time format
- New: Made surgeweb customisation interface in surgemail admin page slightly more obvious whether global / group / domain settings are being edited
- Fix: Messages automatically copied to Sent finally have right time! (correction: still broken on some systems)

Wed 9 April 2009 : Beta build version 4.0j-7

- New: Full support of UTF-8 character handling throughout email addressing, email sending, and contact handling - I think I have everywhere anyway... Should also mostly correct contact entries that were non ascii and previously stored in the contacts database.
- New: Keep <pre> blocks within message correctly formatted - both for display and for sending
- New: Horizontal scrollbar displayed to full message display if the message content warrants it (eg wide pre sections or wide images)
- New: Not directly surgeweb, but surgemail now defaults to using friends in list mode => means surgeweb spam handling display ends up in one of the 3 recommended configurations on default install
- Fix: Messages copied to Sent folder were "out by one hour"... attempted fix but last minute testing it's worse now :-(
- Fix: Certain contacts were resulting in an invalid index character which resulted in contact list completely failing to load
- Fix: Autologin to user.cgi spam page was still broken in IE
- Fix: Autologin to surgeplus filesharing ended up in surgeplus calendar

Sat 4 April 2009 : Beta build version 4.0j-4

- Fix: Use of absolute positioning in styling message elements could leave message parts "all over the surgeweb interface".
- Fix: Autologin was not working for some username and password characters

Wed 1 April 2009 : Beta build version 4.0j-2

- New: More informational warning message when a message fails to send
- New: Context menu on recipients when composing messages (allows: remove, add as contact, send new message)
- New: Clicking recipients composes a message to the recipient rather than trying to add the recipient as a contact
- New: Allow many recipients on one message all be displayed instead of summarised
- New: Allow many attachments on one message all be displayed instead of summarised
- New: Make comma and semicolon characters behave like tab when it comes to address autocompletion
- Fix: Make address autocompletion work again in the Basic interface
- Fix: Allow images on popup windows was no longer working if the server response was very fast
- Fix: Make address autocompletion work again on basic interface
- Fix: Nasty bug: If a message was saved as a draft before the first attachment was added the message would be sent without attachments even though a cursory inspection showed attachments to be fully attached.
- Fix: Addresses added client side it the contact list again when sending

Tue 31 March 2009 : Beta build version 4.0i-1

- Fix: No longer create duplicate contact entries browser side when sending messages. !OOPS! now does not add them client side at all on message send - too late for 4.0i-1 but will be fixed with next build
- Fix: First ever email "Send" on account whilst surgeweb addressbook is empty does not fail with "NO Found 0 matches" error.

Mon 30 March 2009 : Prerelease build version 4.0h-8

- New: "crossdomain" logins can be disabled / enabled. ie setup customisation etc for say domain2 and then login with user@domain1 to get domain1 look and feel
- New: Ability to customise whether domain is displayed on the login page or not
- New: Ability to login to surgeweb using any vdomain using single url (note that means customisation of default domains is used for any domain that uses this)
- New: Warnings on surgeweb customisation page of admin interface for: customisation of non default vdomain, and group customisation
- New: Autologin support to autologin into surgeweb
- New: Autologin connect to user.cgi / surgeplus - means user.cgi pages to backend servers autologin; and means user.cgi pages continue to work even if someone else logs in to user.cgi from elsewhere
- New: Having IP addresses and / or multiple ports specified in g_webmail_port etc now works with forcing to https and retrieving summary user.cgi information filtering and spam control in options

- New: 'Save message' available from message more actions menu to download message as text file to save it locally
- New: Further modification to the simplified spam configuration interface

Mon 23 March 2009 : Prerelease build version 4.0h-3

- New: The simplified surgeweb Spam configuration interface fully operational
- New: The already implemented keyboard shortcuts documented in the preferences
- New: Which buttons get displayed in the main surgeweb toolbar can now be customised and the default set has been reduced
- New: Always allow images from this sender now operational :-)
- Fix: Serverside enumeration of addresses when sending to group works again (broken mid January with enhanced addressbook fields)
- Fix: Tab key behaves again like a tab key for safari OSX
- Fix: On OSX meta key behaves as it should (most notably apple-N opens new browser windows rather than new surgeweb message)

Thu 19 March 2009 : Beta build version 4.0g-1

- Fix: When downloading multiple messages during caching, charset of iconv conversion of first message to require conversion to utf-8 would be used on all messages that required conversion

Wed 18 March 2009 : Prerelease build version 4.0f-3

- New: Contacts csv import "add unrecognised data to comment" is now optional
- Fix: Incorrect grammar in terms of Login buttons etc
- Fix: Spam button always displayed for messages in spam folder (even if spam rating is low)
- Fix: If messages older than the most recent 100 were deleted/moved with a non surgeweb email client, surgeweb would still display the message in the message list and display a blank message if you clicked it
- Fix: Addressbook import results in contacts functionality failing with ajax error, fixed but invalidly formatted contacts may have a warning about missing information
- Fix: IE "un terminated string" error if there were double quotes around name part of address - was not causing any actual problem though
- Fix: Correct display of CID inline images missing a CID tag
- Fix: Correct display of messages with X-NotAscii header different from mime content header
- Fix: Special characters (single quote in particular causing problems) in attachments broken both for message display as well as attachment uploading

Fri 14 March 2009 : Prerelease build version 4.0d-7

- Fix: Solaris_x86 crash on every surgeweb login fix (may have affected some solaris sparc versions too)
- Fix: Crash on solaris sparc messages that needed ICONV translation

Mon 9 March 2009 : Prerelease build version 4.0d-2

- New: Version check that disables surgeweb interface if surgemail binary is incompatible with surgeweb interface template files. (eg. after surgemail binary only upgrade)
- New: Ability to report Spam / NotSpam from various locations in the interface. This combines multiple user.cgi actions including the new "cleaning of stars" from false positives - for more information see: [surgeweb spam marking](#) - and requires g_imap_friends "true" in order to be enabled.
- New: Purge button next to Spam folder
- New: "Friends Pending mail store" displayed as "Spam folder" within top level folders if g_imap_friends enabled, else if any ordinary folder named Spam exists - display in same place, else display no spam folder at all.

Fri 6 March 2009 : Beta build version 4.0c-1

- Fix: Several fixes indirectly related to surgeweb (fixes to use of g_imap_friends and surgeplus image sharing)

Thu 5 March 2009 : Prerelease build version 4.0b-20

- New: Support for SSL IMAP and SMTP connections between surgeweb and backend server
- New: Support for separate backend IMAP and SMTP servers
- New: Support for non surgemail backend IMAP servers (tested against gmail's IMAP servers) - note surgemail only features are not yet getting disabled if non surgemail IMAP servers are used, and user.cgi/contacts does not work/is not yet stored.
- New: Prevent back button from killing messages you are editing
- Fix: Make tab and enter keys work as requested when composing message (enter goes to next field, tab in editor indents - tested in mainstream browsers)

- Fix: No longer switch to login page if Enter pressed in Subject field when composing a message
- Fix: Showing preview was no longer scrolling preview window to top - broken by new message caching code
- Fix: Reply all was not including any cc recipients in original message - broken by new message caching code

Fri 27 February 2009 : Prerelease build version 4.0b-19

- Fix: Display of messages that had a cc field was slightly wrong, most noticeable by red box javascript error if one of these messages was at top of the list when displaying a folder - broken by new message caching code

Wed 25 February 2009 : Prerelease build version 4.0b-18

- Fix: Handling of ' character in recipient addresses - broken by new message caching code
- Fix: Truncation of recipient list if list >1KB (approx 35 addresses) - bug already existed, but made more visible by new message caching code
- Fix: Bug > 4 recipients - bug already existed, but made more visible by new message caching code
- Fix: Really long attachment filenames error - bug already existed, but made more visible by new message caching code
- Fix: Show more error information in some cases which asked to report error code of []

Mon 23 February 2009 : Prerelease build version 4.0b-17

- New: Major rewrite of the message handling & caching - these are important changes and have been quite well tested but doing this _may_ have broken just about anything in surgeweb :-(
- Fix: Bug with variety of symptoms including "msg_show prerequisites not met" when viewing message, and other javascript errors (most likely to occur when logging in)
- Fix: Resent-from header getting displayed on messages that it should not be getting displayed on
- Fix: Mixing existing message attachments (eg fwd attach or forward) with newly uploaded attachments did not work
- Fix: Popping messages in and out of the main window now keeps attachments
- Fix: Edit message window comes up with correct layout immediately (previously it displayed redirect layout until message was ready for display)
- Fix: Occasionally spaces between text were getting eaten for some html messages
- Fix: Some folder and message id confusion resolved (eg previously if you opened 2 messages in a tab, then replied to the first you would end up with a reply to the second message)
- Fix: Odd behaviour of prev next in message display windows fixed if these are clicked after having switched folders

Fri 20 February 2009 : Prerelease build version 4.0b-16

- Fix: Running a spellcheck sometimes concatenated some of the lines in the original message to/from/date headers.
- Fix: Sending using Opera frequently "totally messed up" email messages by truncating the body in a variety of odd ways.
- Fix: Actual address autocompletion (ie pressing tab/enter during autocompletion) was not working in opera.

Tue 17 February 2009 : Prerelease build version 4.0b-15

- New: Surgeweb customisation files now get mirrored as you modify them with the web admin interface (not during resyncs)
- New: Warning telling users to save their work and relogin if users are in an ajax surgeweb session when surgemail gets upgraded
- New: Spellchecker "Learn" works saving words in a users personal dictionary
- Fix: Some files were not getting fully refreshed after surgemail upgrade if "Remember me" tickbox was used to login
- Fix: Basic template does not correctly disable surgeplus / blogs features
- Fix: Username on login is no longer case sensitive resulting in different preferences
- Fix: Queuing changed so it is now impossible to get half broken interface with permanently scrolling info icon - happened if there was any javascript error processing ajax responses, now will always report an error to user and other ajax requests continue to work

Mon 16 February 2009 : Prerelease build version 4.0b-14

- Fix: Minor tweaks to several of the fixes in 4.0b-13

Mon 16 February 2009 : Prerelease build version 4.0b-13

- New: "Drag and drop + hold for 1s" opens a collapsed folder for the duration of the drag and drop
- New: Consolidated selection cursor coloring throughout the interface
- New: Improved the folder left panel and folder menu in terms folder "clickable/droppable hot area" spans the full width of the panel, and improved selected folder display
- Fix: Spellchecker was failing at the first character in Safari or Chrome

- Fix: Spellchecker menu was not correctly placed if message window was scrolled
- Fix: selecting next / prev on popup message windows was bringing main window to front frequently in IE

Thu 12 February 2009 : Prerelease build version 4.0b-12

- New: Much nicer "updates handling" when editing contact information. Includes variety of changes and specific bugfixes mentioned below.
- New: "Send >>Done" button also appears on other "normal folders" other than inbox
- Fix: Contacts - File by company tickbox sometimes needed to be clicked twice
- Fix: Contacts - Saving contacts resulted in duplicate in memory contacts list
- Fix: Contacts - Search for Contacts page and contacts picker always searches email, firstname, lastname, and company.
- Fix: Improved the recent fix to drag and drop handling (new symptoms were stop sign cursor in safari after scrolling message list)

Tue 10 February 2009 : Prerelease build version 4.0b-11

- New: Spell checker
- New: From addressing "locked down" to avoid easy faking of emails (for now manually add address in "allowed_from" setting in _user.dat if from address must be changed, feature planned here)
- New: Move to done button on the main menu bar
- New: Make surgeweb work with g_url_alias and url_alias settings
- Fix: Messages sent with basic template could be sent with no email address in from header
- Fix: Safari & chrome no cursor displayed in blank editor window issue
- Fix: Messages not displaying "for a while" whilst clicking folder (happened when some action had been taken on that folder in the last 15s - eg. sent message saved)
- Fix: When using select "all in folder" feature the drag and drop message count correctly displayed
- Fix: On login the inbox unread message count is immediately correct (previously correct on any subsequent web request)
- Fix: Pressing reply etc on popup message reading windows was not working (broken about a week ago)

Sat 31 January 2009 : Prerelease build version 4.0b-10

- New: https_require available on customisation screen to force browsers into using HTTPS for security
- New: logout_url available on customisation screen to specify the page you go to when pressing the logout button
- New: help_url available on customisation screen to the base location for customising the help files (actual help file names hardcoded appropriately in interface eg. ajax.htm, basic.htm, mobile.htm, user.htm)

Fri 30 January 2009 : Prerelease build version 4.0b-9

- New: Further selection handling improvements (to avoid accidental message deletion when using tickboxes, to cope with auto refresh, and to make drag & drop nicer)
- New: Several improvements to message drag and drop handling
- Fix: Contact drag and drop into groups was not working in Firefox

Thu 29 January 2009 : Prerelease build version 4.0b-8

- New: First autoselected message is a "fake selection" and more like a cursor - autodisplays first message in inbox, but means it cannot be accidentally deleted without explicitly selecting the message
- New: Auto refresh - When new messages arrive the message list is automatically refreshed with them (provided inbox is displayed, and you are not search limiting or dealing with a multi-item selection)
- New: Auto refresh - If you started with 100 messages on a page and had more than 100 in the folder, when you drop below 50 in displayed list (eg deleting or moving messages) it refreshes the list automatically to display up to 100 messages again
- Fix: Deleting / moving the last message in a folder clears the preview pane

Wed 28 January 2009 : Prerelease build version 4.0b-7

- Fix: IE no longer gives a "display insecure content" warning on https connections
- Fix: Pressing close on popup message window displayed navigate away from page warning dialog
- Fix: Folders with ' character (single quote) result in surgeweb ajax hanging on login - now properly handled throughout the interface (hopefully everywhere ;-)
- Fix: Use of & character in reply name truncated reply name
- Fix: Mouseover hints over editor toolbar appearing again
- Fix: Clicking search and pressing ">" or ">>" shows next messages in search not folder
- Fix: Reply directly without clicking a message first (ie just selecting tickbox) resulted in error and unusable interface - look at scroller (also

affected other message actions)

- Fix: Forward/forward attach of multiple messages only attached one message in popup mode (it was attaching all messages in tabbed mode)
- Fix: Surgemail crash if message deletion attempted on session which is no longer valid (eg you are mid session and surgemail is upgraded or restarted, and you press delete before surgeweb checks for new messages)

Wed 28 January 2009 : Prerelease build version 4.0b-5

- Fix: Shared contact warning displayed when creating new contacts in personal addressbook
- Fix: Additional instances of contact information gets created when sending to addressbook entries in "old style" addressbook formatting
- Fix: Crash when trying to re-add already deleted contacts

Tue 27 January 2009 : Prerelease build version 4.0b-4

- New: Significant further development of the contacts editing interface (a lot has needed to be changed, some things that were working may be broken until fully tested over the next few days)
- New: Shared addressbooks can be edited using the contacts editing interface (provided access_group based write access has been granted using abook surgemail.ini setting)
- New: Import of outlook formatted cvs addressbook files
- New: Additional webmail addressbook fields now displayed within surgeweb
- Fix: Contact related unusual character encoding issues resolved

Mon 12 January 2009 : Prerelease build version 4.0b-2

- New: Contact information editing interface (includes manual contact creation, large variety of arbitrary fields, smarts like address field links to google maps)
- New: Contacts editing letter index works now
- New: Warning on contacts editing page if no contacts data source is ticked
- Fix: Handle capitalised letters in the "surgeweb" part of the url

Tue 5 January 2009 : Beta build version 4.0a-1 (Full surgemail beta aimed to take to release status)

- Fix: Drafts folders displayed the same way as Sent with the "to addressing"

Tue 30 December 2008 : Beta build version 3.9h-74

- Fix: First message row behaved oddly for some shift selection handling
- Fix: Make clicking on the ajax address autocompletion list fill in address correctly
- Fix: Make further interface customisation work for other special folders (eg purge for "Deleted Items" or "Deleted Messages"
- Fix: If non default "Sent" folder is used (ie "Sent Items" or "Sent Messages"), display the to header instead of from header correctly
- Fix: Select "all messages in folder" feature gets disabled as soon as searching is applied
- New: Sending message with 'bcc' but blank 'to' adds 'To:(Recipient List Suppressed)' header
- Fix: Make Mobile and Basic interfaces default to reverse date again (broken with message list sorting implementation last week)

Tue 30 December 2008 : Beta build version 3.9h-72

- New: Documentation in the online help on the search facilities (accessible from the dropdown menu in search field)
- Fix: HTML menubar works in basic html template
- Fix: IE sometimes messed up the rendering of the buttons in popup compose windows
- Fix: Editing messages in popup windows (broken with font size fix this morning)

Tue 30 December 2008 : Beta build version 3.9h-71

- Fix: Sending messages in text format (from mobile interface) was incorrectly adding the "This message is multipart mime" line
- Fix: Editor font names streamlined to use what seems to be more common fonts + have fallback fonts if the actual font is missing
- Fix: Editor font sizes streamlined and can now set default message font size in config files
- Fix: Filtering & Spam control summary page status was not getting displayed under some circumstances
- Fix: In IE full dd/mm/yyyy display in message list does not get hidden slightly on right
- Fix: Streamlined titlebar display (\$domain\$ / \$account\$ no longer required in custom/config_*.dat files)

Wed 24 December 2008 : Beta build version 3.9h-70

- New: Display menu which displays 'message subsets' using server side folder searching mechanism (eg read/unread/replied/unreplied/last

day/last week)

- New: Serverside 'single folder' search implemented
- New: Sorting message list implemented (by date/to/from/subject/size)
- New: Surgeweb native addressbooks get mirrored
- New: Surgeweb stores user preferences in "_user.dat" (instead of "user.dat") and file gets uploaded to imap server (and thus mirrored)
- New: Surgeweb can connect to backend surgemail imap server (g_surgeweb_backend_server, surgeweb_backend_server, surgeweb_backend_web)

Mon 22 December 2008 : Prerelease build version 3.9h-68

- Fix: In folders with non cached messages rapid navigation with arrow keys could cache message incorrectly (mixing up messages)
- Fix: Further fix to 'indefinite scroller' when sending
- Fix: Re-editing drafts keeps addressing again (broken with addition of surgeweb native addressbook handling)

Sat 20 December 2008 : Prerelease build version 3.9h-67

- New: Ability to customise the special folder list eg to add say "Spam" or "Archive" to Inbox/Drafts/Sent/Trash
- New: Include configuration warnings if user.cgi "Add all outgoing email addresses to whitelist" is not set
- New: Whole email address displayed in browser titlebar (as opposed to just domain)
- Fix: If tabbed message displayed and different message selected in message list after opening tabbed message, message menubar actions (reply / delete / move etc) would apply to wrong message (ie the newly selected message in message list)
- Fix: 'Please select at least one message' dialog was displayed if moving a message from tabbed message display with 'select next message' set to none.
- Fix: If Flash based send sound failed for some reason the sending channel remained locked and the scroller would continue indefinitely
- Fix: Made enter key when creating folders work properly
- Fix: 'To' button contacts picker needed two clicks for popup messages on non IE based browsers

Thu 18 December 2008 : Prerelease build version 3.9h-66

- New: 'Filtering & Spam Control' access to user.cgi functionality fully functional

Tue 16 December 2008 : Prerelease build version 3.9h-65

- Fix: Variety of little tidypups to remove odd behaviours from the contact management page
- New: Email contacts directly from the contacts management page
- New: Additional warning when trying to organise shared or webmail contacts on contacts management page
- New: Disable fancy addressing preference - disables the fancy address fields (and address picker) - probably just useful debugging
- New: Sticky menus preference - advanced options preference to make menus hang around until click is explicitly received before disappearing
- Fix: Old browser compatibility improvements for Netscape browsers (tested using netscape 7 - html and basics of Ajax work)
- Fix: Made message move to folder and message copy to folder work on html interface

Sat 13 December 2008 : Prerelease build version 3.9h-64

- New: surgeweb.log rollover size also set by g_log_size
- Fix: Caching fix for really long links in html messages
- Fix: Further attachment order fix for unix systems
- Fix: Made attachment upload work on Safari and Chrome again (broken with changes a few days back)

Wed 10 December 2008 : Prerelease build version 3.9h-63

- Fix: Further fix to make some webmail addressbooks work correctly + display spaces in webmail addressbook name nicely
- Fix: When attaching more 10 images in one email the filenames were being prefixed with '_' character, and attachment order was messed up
- Fix: Filtering and spam control dialogs close button works in firefox now
- Fix: Improve tabs display when using "panelised" customisation mode

Tue 9 December 2008 : Prerelease build version 3.9h-62

- New: Quite a lot of improvement to the contact management page (group creation, group deletion, prettified page & tied to tailorable left column)
- New: Made left column "Mail" pane tailorable to each application level menu item

- Fix: Further fixes related to special character handling
- Fix: Support sending to contact group names with spaces

Mon 8 December 2008 : Prerelease build version 3.9h-61

- Fix: Variety of fixes related to special characters not working (eg '&' in addressing whilst sending messages)
- Fix: Fancy addressing sometimes introduced a blank address (eg on popout)
- Fix: Compose via more actions dropdown was not working yet
- Fix: Message html edit menus had recently been broken
- Fix: Interface elements hidden in response to g_user_access (in addition to the user_access authentication database field it was already using)

Tue 2 December 2008 : Prerelease build version 3.9h-60

- Fix: Nasty little bug in the new native contact handling that would "break the whole surgeweb interface" in several ways, if you had webmail addressbook files present with certain formatting

Mon 1 December 2008 : Prerelease build version 3.9h-59

- New: Drag and drop based contact organisation into groups
- New: Drag and drop based message moving and copying

Fri 28 November 2008 : Prerelease build version 3.9h-58

- New: Bunch of further implementation on the native addressbook handling
- Fix: Further fix to the "reply_all" functionality (I think it finally fully works again now)

Web 26 November 2008 : Prerelease build version 3.9h-56

- New: Addressbook handling native to surgeweb supporting - addresspicker, shared addressbooks (global/domain/g_access_group based), distribution list type group handling, fancy to/from/cc addressing fields, surgeweb native addressbook management page (STILL INCOMPLETE)
- New: Surgeplus menu removed for surgeplus features in iframe (just like old webmail)
- New: Tab handling improved (name text and formatting - still needs resizing if there are too many tabs)
- New: Access to user.cgi pages in a dialog window from options page
- New: Summary spam control page (from Options - Filtering and Spam control) (STILL INCOMPLETE)
- Fix: CC header incorrectly output as To when sending messages (recently broken, surgeweb symptoms reply-all does not work on messages like this received)
- Fix: Relogin dialog captures keyboard events - ie can press enter and does not do things on background page

Fri 14 November 2008 : Beta build version 3.9h-54

- New: HTML editor enabled for signature editing
- Change: Make redirect work like other mail clients (keep to/from/reply-to intact and add additional headers) & display Resent-from header
- New: Ability to display arbitrary additional headers in message display
- Fix: Replied indicator was not getting updated when messages are replied to (broken with mondays additions)
- Fix: Sending message sometimes resulted in "Please select at least one message" error (broken with mondays additions)
- Fix: Properly web encode subject line in message lists (allows >, < characters be displayed correctly)

Mon 10 November 2008 : Beta build version 3.9h-53

- New: Surgeweb makes use of Reply-to headers when replying to messages
- New: 'Copy to folder...' capability as well has 'Move to folder...'
- Change: Somewhat changed "Move to..." (folder) implementation and made "Move to..." work for popup messages
- Change: Reply to all, no longer include ones own email address in list of recipients
- Fix: No cookies warning on numeric urls (eg. http://127.0.0.1/surgeweb - affects some browsers only: google chrome & konqueror that I'm aware of)
- Fix: Surgemail crash when trying to move folders under Konqueror (move still fails, error displayed instead)
- Fix: "Send & move to done" moves all attached messages to Done folder for "Forward (attach)" sending
- Fix: Made the auto "select next/previous" work on send too (just like existing delete & move behaviour)
- Fix: Allow " characters in signature and reply address without it "breaking the ajax interface"
- Fix: Click on text and html message parts displays directly again (rather than showing default_attachment_name dialog)

- Fix: Mobile interface was showing characters in password field

Thu 6 November 2008 : Beta build version 3.9h-52

- Fix: Administrator interface help links point at the online admin documentation
- Fix: 'Beta Testers' pane now points at online documentation
- Fix: Made admin interface work under Firefox again (recently broken with new feature addition)
- Fix: Downloads of attachments with spaces in the name was failing to keep the full filename
- Fix: Subfolder folder sorting issue fixed (this was tried 2 days ago too but first fix only created worse problem)

Web 5 November 2008

- New: Basics of online admin documentation written
- Fix: Several minor fixes

Mon 3 November 2008 : Prerelease build

- New: Handle "iso-8859-1" as "windows-1252" to correct mailers that generate invalid encodings
- New: Message printing capability
- New: No cookie warning on urls without a "." character
- New: Admin interface improvements
- New: Easy to configure custom 3rd column content
- Fix: Correctly display attached messages of type message/rfc822 without a filename
- Fix: Surgeweb no longer "eating" space characters in some messages

Sun 2 November 2008 : Prerelease build

- New: Optional automatic image downsizing
- New: Size information displayed with uploaded attachments
- Fix: Support for Flash plugins v10+ for attachment uploading
- Fix: Additional space between signature and original message
- Fix: Variety of minor fixes

Fri 31 October 2008 : Prerelease build

- Fix: Setting "Message list - select none" means no messages are autoselected on login or folder switch
- Fix: Selected folder highlighting in black works in all browsers now
- Fix: Move to menu scrolls to top each time you use it

Thu 30 October 2008 : Prerelease build

- Fix: Variety of fixes
- New: Sending format improvements - correctly handling non ascii charactersets as utf-8 & nicer message content formatting
- New: Support for third column with advertising content etc
- New: More admin interface functionality

Fri 24 October 2008 : Prerelease build

- New: Basic HTML interface functional and "mostly complete"
- New: Mobile interface functional and "mostly complete"
- New: Normal forward of multiple messages switches to forward attach (with multiple messages)

Fri 17 October 2008 : Prerelease build

- New: Forwarding keeps attachments
- New: Forward (attach) - ie forward with original message as attachment
- New: Redirect - Send on original message as intact as one can
- New: Edit drafts keeping attachments

Wed 15 October 2008 : Prerelease build

- New: Variety of new settings to configure
- New: Surgemail admin interface for customisation
- New: Support for group based customisation

- New: Hierarchical 'permanent caching' mechanism that allows parts of the userbase browser caches to be invalidated

Fri 10 October 2008 : Prerelease build

- Fix: Improved positioning of message popup windows
- Fix: Made surgeweb handle '.' characters at the start of line correctly when sending via smtp
- Fix: Have 2 blank lines when replying to message (broken yesterday)

Thu 9 October 2008 : Prerelease build

- New: Work on html template (still incomplete) and able to switch to html from ajax
- New: Improved customisation support (continuing to be worked on)
- New: Made panelisation fully work and changed the default look slightly
- New: More efficient webmail temp files and improved temp file removal
- Fix: Variety of minor fixes & improvements

Tue 7 October 2008 : Prerelease build

- Fix: Got customisation all setup to be "usable"
- Fix: Made surgeweb drafts editable in other clients (like webmail)
- Fix: Fixed bug that was interaction of mirroring with webmail trash clearing making surgeweb message deletion not work sometimes

Mon 6 October 2008 : Prerelease build

- Fix: Various minor fixes
- New: Cache newly received inbox messages, and don't treat copied messages as new
- New: Improved / more sensible / fixed message selection behaviours
- New: More efficient message caching and tidying up of cached files
- New: Appropriately modify the toolbar based on which folder on is in
- New: Basics of addressbook integration (autocompletion without using surgeplus, click to add contact to addressbook links, autoadd people you sent mail to addressbook feature)
- Fix: Send was _sporadically_ losing session information and asking to relogin (fixed, but if session is actually invalid, eg surgemail restarted, you will still be prompted for relogin)
- Fix: Made switching folders with background actions pending and not yet complete safer

Fri 3 October 2008 : Prerelease build

- New: Opening messages drafts folder opens them for editing (still incomplete)
- Fix: Keyboard commands work whilst in editor in FF
- New: Message drafts autosaving in the background as you compose messages
- New: Purge button on the trash
- Fix: Only have "send & move to done" button appear if message was in inbox
- Fix: Switching between messages always scrolls to the top
- Fix: Made surgeweb correctly handle links in html emails with "base" tag
- Fix: Several minor improvements including crash fixed

Tue 30 September 2008 : Prerelease build

- New: ReplyAll actually replies to all the recipients
- Fix: Fixed several buffer truncation based issues
- Fix: Display large numbers of recipients in a sensibly truncated form
- Fix: Allow date / time formatting to be specified
- Fix: Variety of little interface "tweaks"
- Fix: Made prev / next message "work" under all circumstances

Sat 27 September 2008 : Prerelease build

- Fix: Entering text on "modal dialogs" was passing commands to message list page (introduced yesterday)
- Fix: More efficient background "check of new mail"

Fri 26 September 2008 : Prerelease build

- New: Allow the selection and manipulation of "all messages in folder" for folders with multiple pages of messages (after all messages on

page selected)

- New: Tooltips on most buttons and improved keyboard shortcuts for many commands (key combination generally shown in tooltip)
- New: Message count on top right of page gets updated with message deletion / move
- New: "Not yet implemented" commands are hidden by default (most anyway :-)

Thu 25 September 2008 : Prerelease build

- Fix: Fixes in handling of text message parts

Wed 24 September 2008 : Prerelease build

- Fix: Further fixes to display of html messages

Tue 23 September 2008 : Prerelease build

- New: Message CSS gets washed - should eliminate message content affecting screen layout
- New: Preferences get applied without manual refresh
- Fix: New message indicator does not show "lots" of new message for first ever login

Thu 18 September 2008 : Prerelease build

- New: Improved keyboard command handling
- Fix: When selecting a message through any form of click, mark read immediately

Wed 17 September 2008 : Prerelease build

- New: Folder management made "nicer" and some bugs fixed
- Fix: A bunch of "little tweaks" throughout the interface

Tue 16 September 2008 : Prerelease build

- Fix: Allow images this time (broken by yesterdays new features)
- Fix: Attachments have size information displayed (various places in ui and during download)
- Fix: Further improvements to the "new mail" counters

Mon 15 September 2008 : Prerelease build

- New: Background "new mail" check that updates "unread counter" and audio feedback (probably still need to do some tweaking)
- New: Unread message count displayed for all folders (some conditions not yet updated)
- New: Message count and unread message count displayed for all folders on folder manage page
- Fix: Require display for 3 seconds before marking message "read"
- Fix: Selection box clicking works again (broken 2 days ago)
- New: Pressing escape while message displayed in tab closes tab
- New: Pressing enter on message opens it (same as double click)
- Fix: Web mode message opening works properly in all browsers now
- Fix: Shift and control selections on message text no longer open blank panes in FF
- Fix: Attachment download links work in "non inbox" folders too
- Fix: Move menu width same as Mail folder list width
- Fix: Made tab order on compose page sensible

Wed 27 August 2008 : Prerelease build

- Fix: Made menus on html editor not "hang around" too long (html editor still incomplete)
- Fix: Make sure that email addresses with ',' and ';' characters in the "quoted part" work correctly
- Fix: Using caching, "cc fields" were getting copied to subsequent emails
- New: For display, surgeweb "flows" the first level of format=flowed text emails received
- New: Surgeweb sends text part in "quoted-printable" instead of format flowed, and correctly limits width of lines in generated email
- Fix: Text part generated from Safari sent surgeweb messages now no longer has all paragraphs concatenated together
- Fix: Webmail will now show the html part of surgeweb messages correctly

Tue 26 August 2008 : Prerelease build

- New: Html editor more (but not yet fully) complete
- New: Save draft working (with attachments, but cannot start editing the saved draft again yet)

- Fix: Occasional subject lines messed up (broken on Friday)
- Fix: Fixes in the highlighting of various menus

Sat 23 August 2008 : Prerelease build

- Fix: Closing individual tabs was not restoring main pane in FF / Safari
- Fix: Autocompletion was not working in Firefox anymore
- Fix: Alert warning if Flash is not available when trying to attach

Fri 22 August 2008 : Prerelease build

- New: CC and BCC fields operational
- New: Display addressing information for messages with multiple recipients nicely (both to / cc fields)
- Fix: Message selection improvements
- Fix: Several fixes to attachments feature

Thu 21 August 2008 : Prerelease build

- New: Attachments fully operational
- New: Download "all attachments" as single zip file
- Fix: Various little fixes

Tue 19 August 2008 : Prerelease build

- New: Validity checked folder names (for invalid slashes)
- Fix: Login from blank work directory messed up screen layout
- Fix: Message compose edit display yellow (broken yesterday)

Mon 18 August 2008 : Prerelease build

- New: Client side UI implemented for attachments - STILL INCOMPLETE. Currently, uploads files to server (surgeweb/work/* currently), but does not actually do anything with these yet... (like attach these to messages!)
- Fix: Fixed issue where going to login page was sometimes blank (previously a refresh already fixed it)
- Fix: Made Web mode behave better in terms of screen resizing.
- Fix: Made preferences & folder management pages behave nicely for server down / session no longer valid situations.
- Fix: "blank" compose / edit pane problem

Fri 15 August 2008 : Prerelease build

- New: "Prettified" the options and manage folders pages
- New: Addressbook autocompletion functional, but integration is rudimentary and still needs fuller implementation
- Fix: Tab switching between messages behaved oddly
- Fix: Sending of message whilst session was invalid resulted in message disappearing entirely, and displaying it had been successfully sent (Now it shows relogin dialog, and allows resend)

Wed 13 August 2008 : Prerelease build

- New: Add date header (+several other headers that should have been there) for outbound mail
- Fix: Display of message date order sorting was broken in Sent folder
- Fix: Using g_imap_old surgeweb would sometimes display folders empty, fixed

Tue 12 August 2008 : Prerelease build

- Fix: Allow surgeweb to be used with g_spam_check_auth
- Fix: Variety minor display & layout fixes
- Fix: Further improvement in display of htm emails & processing of links in html emails
- New: Allow caching of selected messages in arbitrary folders. Only the most recent 500 messages to be cached are remembered.
- Fix: Minor fix to make sure inline & tabbed messages can be displayed without server request (same as popup)
- Fix: Made shift arrow key region selection behave as it should

Mon 11 August 2008 : Prerelease build

- New: Basic folder management (create/delete/rename/move) implemented (still to do display more message information for each folder on this page)

- New: "Done" folder created if it does not exist when "Send ->Done" pressed
- Fix: Lose logout buttons off top of screen sometimes in application mode
- Fix: Edit window display for tabbed/inline compose/edit in safari/firefox
- Fix: Error when setting "real IP address" now logs error but does not prevent message sending

Sun 10 August 2008 : Prerelease build

- Fixed nasty crash introduced two days ago (typically after message delete)

Sat 9 August 2008 : Prerelease build

- Made washing of text emails actually turn various types of urls into links
- Further tidying of HTML washing for display of messages
- Using SMTP authentication when sending
- Added original To: recipients to the "original message" header in the reply/fwds
- Made html editor font defaults more consistent
- Fixed message "move" menu (broken with yesterdays enhancements)
- Upgrade install now installs surgeweb/custom "customisation directory" if it does not already exist
- Added README.TXT files to surgeweb/tpl & surgeweb/custom directories to explain their use

Fri 8 August 2008 : Prerelease build

- Fixed lots of minor annoyances - in particular related to focus, selection, menu behaviour etc

Thu 7 August 2008 : Prerelease build

- Save layout preferences in Ajax interface when they get changed (My Folder collapsed state, pane split sizes)
- Implemented large part of the Html interface (still not functional enough to use as an email client though)

Tue 5 August 2008 : Prerelease build

- Improved the handling of the relogin on session invalid dialog (further testing & improvement to go yet though)
- Users real IP address gets logged to surgemail delivery log
- Sending now done via SMTP instead of surgemail internal functions
- Fixed serverside surgeweb memory leaks I'd been ignoring
- Surgeweb honours user_access & g_access_groups (blogs & surgeplus features)
- Initialise user.dat from surgemail authentication database fields - fullname
- Added "To field" to full message display

Mon 28 July 2008 : Prerelease build

- Improved the apparent responsiveness of logins (just means that the "page loading" information is displayed sooner and has progress info)
- Much improved error condition handling
- Made send & move original to done work again
- Made selection handling significantly faster
- Fixed five issues with display "inline" behaviour for web application mode
- Fixed inline pages (Manage/options/surgeplus) sometimes displaying inbox along with options/surgeplus etc.
- Fixed editor actually went to edit mode for inline replies (that you got to straight from main message list)
- Fixed channel compression problem where it would lose chunks of the sent combined.js file
- Fixed edit window popin problem script crash (IE only) which would result in losing the text being edited and perpetually scrolling status scroller

Fri 25 July 2008 : Prerelease build

- Various other fixes and features.
- Made popout / popin of windows work
- Fully implemented: Application (popup windows / or tabbed) and the Web Page modes
- Implemented "remember me" cookie based logins & Made authentication process secure
- Display layout now stored as a preference (& preference handling much tidier)
- Enabled permanent life caching (same mechanism rest of surgemail)
- Made reply work for popup windows & tidied up request handling to 3 request types: foreground (cancelable, queued, background)

- setup message popup windows to handle own connections to the server (mostly will actually pass requests to main window though)
- make delete in popup work in main window (but bit ugly still needs fixing...)
- dblclick message popups display formatted message content
- Request queue implemented (only for used for status flags, delete for now - will add others soon)

Mon 7 July 2008 : Prerelease build

- Double click in app modes opens message in popup (just raw mode for now, will be extended to full window shortly)
- Give left columns (& preview pane) resize handles
- Preview pane displayed vertically / horizontally
- Implemented full screen "application modes"
- Added "Send and move to Done" option

Fri 20 June 2008 : Prerelease build

- Inbox caching resulted in message read status set to read. Not anymore.
- Improved refresh if messages were removed from the server external to surgeweb would sometimes not be removed from surgeweb
- Make inbox caching work properly if you start clicking other folders straight away
- Made multiple concurrent requests for the same message safe serverside
- Fixed variety of issues related to "missing messages" in displayed list - should work MUCH better for large mailboxes.
- web_encode text parts correctly for primary message display
- display the "raw" / "text" / "html" pages in native character set for the part in question
- Optional audio confirmation of successful background send (like imail)
- display multipart/signed messages correctly
- make the display of messages with "blocked images" work with inbox caching
- This history information added (and recorded changes over the last few days added retrospectively)

Tue 17 June 2008 : Prerelease build

- Some messages extra info displayed incorrectly when displayed from inbox cache (due to "images blocked" and "attachment count" etc not getting cleared between invocations to render the message)
- Date display "simplification" around user preferences based timezone rather than GMT
- Read flag gets updated when a message gets read through surgeweb
- Replied / forwarded / new last imap check indicator added on left
- Replied flag automatically gets set when replying to a message
- Listens to surgemail.ini g_surgeweb_work for work directory
- surgeweb.log always created in expected location
- g_surgeweb_debug allows detailed logging for individual users
- improved selection handling

Mon 16 June 2008 : Prerelease build

- Sending automatically copies messages to the sent folder
- Improved to / from address handling and allow name/address to be specified in from
- Fixed crash if mime decoder finds invalid message part
- Sent folder displays "to address" instead of from address
- Crash on login with url with surgeweb url with trailing slash
- Replying to messages adds "Re: " if it does not exist already
- Message copied to sent folder whilst sending gets flagged as "Seen"
- Attachments download links added and image attachments view and download links fixed

Sat 14 June 2008 : Prerelease build

- Sending is now MIME multipart alternative. The text part gets automatically generated from the html, and should be pretty "clean". Let me know if anyone sees anything to the contrary
- External images is now properly handled. Images are blocked by default with the option of allowing them to be shown. The allow is an allow once only (implemented), allow always for sending email address (not yet implemented), or in the options there is preference to allow always

(implemented).

- Display of raw message / text part of the message or html part of the message is now correctly handled (using links top right of the message display)
- Shift based range selection for selecting multiple messages (using the checkboxes) now functional

Fri 13 June 2008 : Prerelease build

- Error handling added during the actual message send - it should now report any errors and allow you to resend etc, whereas previously it would eat the message and happily say "sent ok" :-((
- Fixed the fact that surgeweb would "sometimes" show you old messages that were already deleted on the server (using a non surgeweb client)
- Move messages to folder "menu" added. I'm curious how useful you find this as it is a slightly non standard menu interface.
- Compose and Reply fills in the correct to and subject headers
- Email address with the full in brackets address information displayed when looking at a message
- When done with message switch to X option: list pane (default) / next message / previous message
- Provides status information on the deletion of and moving of messages in the background

Thu 12 June 2008 : Core surgeweb approaching being usable as a mail client, change history recording started

SurgeMail

Simply one of the best; scalable secure carrier class Internet mail server

Fast, Robust, Secure HIPAA Compliant - Mail Server Software

SurgeMail is one of the fastest, most robust and fully-featured secure email server available on the market today supporting all the protocols IMAP/POP3/SMTP and relevant standards. Over 10,000 email servers installed, serving millions of email accounts world wide.

Easy to manage and install

SurgeMail is an advanced secure easy to manage and install high performance email secure server using fast browser based administration tools. Surgemail includes user and domain self administration.

Advanced spam prevention and virus filtering

Surgemail mail server software provides seven level spam and virus protection. The most effective spam prevention techniques are supported including SPF, RBL, challenge-response, white listing and hundreds of other built in spam prevention mechanisms.

Unlimited users and domains

Surgemail is scalable to an unlimited number of users or domains. Surgemail's efficiency mean tens of thousands of users can be supported even on very modest hardware.

Brand new SurgeWeb Ajax / Web2 web email interface

A fast, efficient and customisable webmail interface. Using advanced Web 2.0 / Ajax mechanisms this web interface provides the speed and ease of use normally only seen in a local email client.

RFC Standards Compliant "IMAP Mail Server"!

We take RFC compliance seriously. For example unlike many of our competitors our IMAP mail server module passes the rigorous [unofficial IMAP test suite](#). This means it's more likely to work seamlessly with multiple email clients (IPhones, Android etc...) without 'odd' behaviour.

Linux Mail Server or Windows Mail Server?

SurgeMail email server software was designed to run on both Windows and Linux from day one. It is not a Linux email server product ported to Windows or visa versa, so it works properly in each environment, and you can choose at any time to move from one platform to the other (at no cost) or even run a mirrored cluster of both at once. This flexibility is very valuable as your company policies change you can maintain your email system without painful migrations/conversions. Our goal is to provide the best possible Windows mail server or Linux mail server.

High Performance POP3 Mail Server

SurgeMail's POP3 mail server module is fast, RFC compliant, and totally reliable. And works with the IMAP mail server allowing users to seamlessly switch between IMAP and POP3 or even use both at once. Many normal mail servers cannot permit both protocols to be used at once! (Both IMAP and POP3 modules are standard with all SurgeMail versions)

Windows/Linux SMTP Server Software

The heart of your mail server is the SMTP protocol, SurgeMail's smtp module is designed to handle Incoming Email, Spam, Hackers and Local Users all at once. Many unique features are used to throttle spammers, detect and block hacking attempts, reject the bulk of spam and still accept all normal incoming email quickly and efficiently.

What is a "Secure Mail Server"?

- Designed for security from the ground up, so it can't be exploited as a back door into your computer network.
- Supports SSL protocol for login and smtp so messages and passwords are encrypted at all times and not visibly floating around on the internet for anyone to see/steal (as is common with most servers)
- Includes optional '[SurgeVault](#)' Encryption module for 'end to end' encryption. This module makes SurgeMail



HIPAA compliant!

- SurgeMails unique 'mirror' feature allows live replication of your configuration and mailboxes, this feature will save you your job! A nightly backup is just not a workable solution for your mail system when the most valuable data on your mail server is all 'new and unread' email that arrived within the last day!

RFC Standards Compliant "IMAP Mail Server"!

We take RFC compliance seriously. For example unlike many of our competitors our IMAP mail server module passes the rigorous [unofficial IMAP test suite](#) This means it's more likely to work seamlessly with multiple email clients (IPhones, Android etc...) without 'odd' behaviour.

Groupware support

Optional calendar, file sharing, instant messaging, forum, blogs and chat room modules.

YesImOnline - Free IMAP email client for Windows, Mac, iPhone, IPAD

This handy IMAP email client is easy to use, fast, works with your existing email client or by itself, and makes email more fun and friendly a bit like instant messaging. [Click here to try it.](#)

Unparalleled technical support

Great technical support and a very helpful user community. Read the mailing list for more information, but often features will be implemented and ready to use a matter of days after someone has posted a new feature suggestion!

Select a focus link below for more information.

Integrated Web based email	Anti-Spam and Anti-Virus	SurgeVault Message Encryption
Web based administration (user, domain, server)	Auto Migration from anything	Unique Live Replication
Full SSL secure support for all protocols	SurgeWall firewall	Installs in minutes
Any user database (ODBC, SQL, LDAP)	Unlimited users	Blogging server
Any platform (Windows, Unix, Mac OSX)	Unlimited domains	Archiving / Compliance
Create free Mail Service (Like Hotmail)	Calendar and File Sharing	Free 5 user license
Unparalleled after sales service	User Photo Albums	IPV6 Support

[previous](#) **CUSTOMER COMMENTS:** [next](#)

Wow! surgemail ROCKS!!! :) I have setup surgemail as a bastion host for 5 of our primary domains. We tried every other mail server out there trying to find something that works and that could keep up with our mail throughput.

I'm BLOWN AWAY by just how good Surgemail is! It is processing hundreds of thousands of messages per day and scanning every one for viruses with RAV and giving each one a SmiteCRC spamdetect score. The dual PIII 1Ghz server is only using 1-5% CPU with spikes to 20-30% and only 150MB out of 2GB!!! Amazing.

If you are looking for a reliable and FAST mail server, BUY SURGEMAIL!

-- Robert Boyle - Tellurian Networks - The Ultimate Internet Connection

Fast AJAX web email for existing mail servers!

Use the fast, efficient SurgeWeb web email interface with your existing mail server to allow users accesess to email from anywhere.

Included with SurgeMail Mail Server - at no extra cost.

Unlike many other mail server producers, Netwin includes the following features free of charge

SurgeWeb - including :-

GroupWare

- Calendaring
- Filestore
- Photoshare
- Blogs

Multiple Skins / Templates

Multi Language

The complete, advanced Antispam system with all versions

SPF

- Sender Permitted From

Friends

- Challenge - Response System

SURBL

- Spam URI Realtime Blocklists

Aspam

- Content Filtering

RBL

- Relay Block List

Mailing List Server

Migration Tool to migrate via POP, IMAP etc...

Related Pages

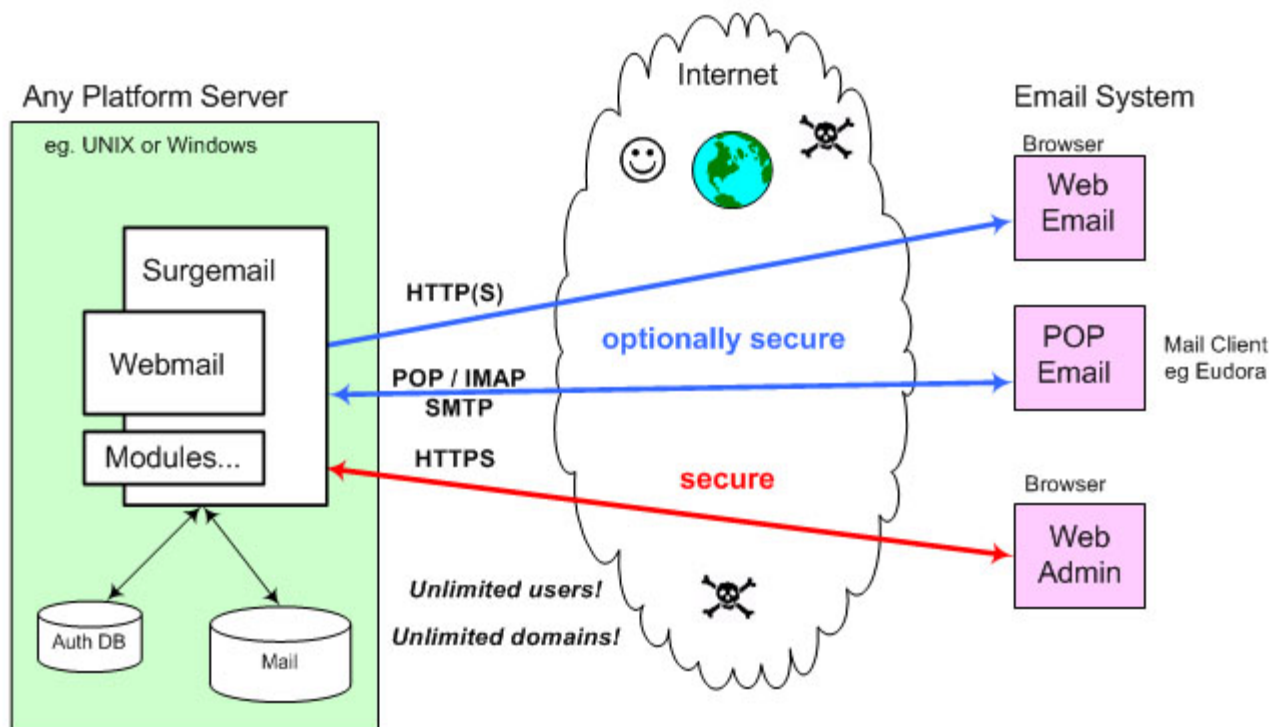
[Surgemail installation guide](#) | [Surgemail manual](#) | [Surgeweb manual](#) | [Email Reputation Database](#)

SurgeMail in a nutshell

SurgeMail is a fully featured enterprise class mail server with integrated web based email, web based account and server administration.

The web based email system builds on NetWins well known WebMail system with extensive customisation options using templates. Combine this with SurgeMails inbuilt web server and you have an all in one solution that does not require extensive work to get components working together.

SurgeMail has many features for high reliability systems such as in-built server mirroring and support for clustered, proxy configurations and NFS based mail storage etc.



Read the following! It will help your understanding of the features available in SurgeMail

How SurgeMail fits together

SurgeMail is a mail server with built in support as a web server. This inbuilt web server functionality allows the mail server to be administered via the [Web Admin Interface](#).

WebMail is a [web based email](#) solution that is served by the SurgeMail webserver. WebMail has its own [Web Admin Interface](#) to manage WebMail only functionality. WebMail communicates to the mail server using standard protocols (POP and SMTP) and could also be served using an alternative web server. If served by SurgeMail WebMail does have enhanced features.

SurgeMail has an in built authentication and user management system (NetAuth for you old timers). This provides web based [account administration for users](#) and [account administration for domain administrator](#). This is again served by SurgeMails in built web server.

Finally an important thing to know is that [SurgeMail Monitor](#) (swatch) is a process that runs separately from your mail server and can be used to restart SurgeMail using the web or monitor SurgeMail for correct operation.

Note: The above links are links to the interface for a default install on localhost. (ie. they **WILL NOT WORK** if you click them and do not have SurgeMail on the system your browser is on!) Copy the URL and edit servename / port number if you have a different configuration.

Setting up a mail server

So you want to setup a mail server, here we will quickly run through what is required and things you will need to setup before actually getting the server up and running. Some of these can be done after the server is installed but it's best to keep them in mind before you start.

Buy your domain name

Generally you will need your own domain e.g. mydomain.com, then you can setup email accounts for that domain, so you will email addresses like user1@mydomain.com etc. People will be able to send to user1@mydomain.com and that user will be able to login with an email client such as outlook express and receive that message, they can also use the built in web based email client WebMail to check their mail.

There are a lot of Internet registrars that sell domains so you should shop around for the best deal you can find and usually buying the domain for a few years at a time gives you a discount.

You can find a list of Internet Registrars from these sites.

<http://www.internic.net/alpha.html>

<http://www.icann.org/registrars/accredited-list.html>

Setup DNS records

Once you have your domain, then you need to setup the DNS records for it. There are several types of DNS records but as far as mail is concerned we are interested in the A records and the MX records.

MX (Mail Exchange) records are only used by mail servers and they tell other mail servers where to send mail for your domain, so it is important to set these up.

If you go to the command prompt in Windows or the shell in *nix you can type
 nslookup
 set type=MX
 netwinsite.com

You will then see this:

Non-authoritative answer:

netwinsite.com MX preference = 25, mail exchanger = smtp.netwin.co.nz

netwinsite.com MX preference = 10, mail exchanger = mail.netwinsite.com

netwinsite.com MX preference = 20, mail exchanger = netwin.co.nz

We can see netwinsite.com has 3 MX records. The way MX records work is the lowest number has the highest priority, so if I send an email to netwinsite.com, the server will first try to send to the the record with 10 (mail.netwinsite.com) if that machine is down, then it tries the next MX record and so on and so on until it finds what that is working and then it delivers the message.

It's a good idea to try and setup at least 2 MX records this of course requires two servers, this means if your main server is down, the mail will get sent to the secondary machine which then forwards it to the primary once it comes back up again. This isn't essential though because even if you only have 1 mail server and 1 mx record, if a server sends to you and your server is down, the sending server usually will keep trying to deliver the message to your server for at least 24 hours.

In the above MX records, each MX record then points at an A record, mail.netwinsite.com is an A record which would resolve to the IP of the machine running SurgeMail.

To setup MX and A records you will need a DNS, this can be a little complicated to setup for some people. Your ISP might be able to help you here, so you should contact them and ask them about this. There are also commercial DNS servers out there, and if you are running a small non commercial server there are some free ones. (<http://www.xname.org>) - just search around on google.

You can also run your own DNS server, on *nix most people run BIND.

On windows you can try "[Simple DNS Plus](#)" which seems very easy and also on windows 2000/2003 server there is a built in DNS server you can use. You definitely need two DNS servers as if you only have one and it goes down for some reason, no one will be able to send mail to you.

* Please note we will not give direct support on configuring DNS servers but can offer advise on setting up DNS records in general.

So, the main things are:

1. Setup at least 1 MX record, so that mail servers know where to send mail for your domain
 2. The MX record needs to point at an A record which has the IP of your server.
- The A record is also what your customers will use in their email clients for the smtp server and pop server.

Example

We have bought our new domain "mydomain.com"

The machine we are going to put SurgeMail on is 1.2.3.4

We setup an A record mail.mydomain.com that points to 1.2.3.4

We then setup an MX record pointing at mail.mydomain.com

Your customers will put mail.mydomain.com in their mail client in the SMTP/POP server sections.

You now need to check that your ISP is not blocking port 25.

At the command prompt type

```
telnet mail.netwinsite.com 25
```

You should get a response like this

```
220 netwinsite.com SurgeSMTP (Version 3.0c2-2) http://surgemail.com
```

If it says unable to connect, then it's possible your ISP is blocking port 25 and you should contact them, or try telnetting again to our server a few minutes later incase we were doing maintenance :). If your ISP is blocking port 25 then all is not lost, you can tell SurgeMail to send it's mail via your ISP's mail server and you can read how to do that [here](#).

You need to make sure that you open all the ports for SurgeMail. You can read which ports are required [here](#). This will require you allowing them in your firewall, and if you are using a router you will need to add pinholes for these ports through to the machine running surgemail.

OK now you have everything ready to [install SurgeMail](#).

After which you will need to do some [post installation](#) steps.

Installing and Upgrading SurgeMail

Windows

To install or upgrade, download the latest version from our website (download link on (<http://netwinsite.com/surgemail/>) or FTP site (<ftp://netwinsite.com/pub/surgemail>) and run it. It is a self extracting archive which will automatically detect your existing surgemail or dmail configuration and upgrade an existing install or perform a clean install as appropriate. You will be asked for confirmation before any action is taken.

If you already have SurgeMail installed and are wanting to upgrade to a new version you just need to download the new distribution and run the installer. The installer will default to upgrading your current version and will leave all your settings as they are. If you have customised your templates then make sure you answer 'NO' to overwriting them when asked which should be the default. SurgeMail will make a backup of these anyway. You can only upgrade to a version that was built within a year from when your license was bought or renewed. It is always a good idea when upgrading to take a backup of SurgeMail if time permits. ([Backup SurgeMail](#))

If you are already running a Mail server it is essential that you stop it and uninstall it before starting SurgeMail as both will not run at once.

UNIX (this includes OSX)

To install or upgrade download the latest version from our FTP site, uncompress and untar the files, then run the install.sh script. The _XXX_ in the file below is the version platform and verison number. (eg surgemail_36f5_linux.tar.gz or surgemail_36f5_freebsd4.tar.gz)

([Or click here to see links to the current release](#))

e.g.

```
ftp ftp.netwinsite.com
user: anonymous
password: anything
FTP> cd pub/surgemail
FTP> ls
FTP> bin
FTP> get surgemail_XXX_linux.tar.gz
FTP> quit
gunzip surgemail_XXX_linux.tar.gz
tar -xvf surgemail_XXX_linux.tar
cd mtemp
./install.sh
```

On a first time installation the standard UNIX mail server will be stopped by commenting out the line in /etc/inetd.conf and sending HUP signal to that process. Also on unix installs you will need to be logged in as root to install surgemail and may need to create the "mail" unix account on the server. In most situations surgemail will create the "mail" account itself if it does not exist.

If you ever need to manually start the server:

On NT based systems (ie XP/2000/2003), type in: net start surgemail

On 95/98, type in: /windows/surgemail

On Unix, /usr/local/surgemail/surgemail_start.sh

If you ever need to manually stop the server:

On NT based system (ie XP/2000/2003), type in: tellmail shutdown or alternatively "net stop shutdown" can be used but tellmail shutdown will provide a cleaner shutdown

On 95/98, type in: tellmail shutdown

On Unix, /usr/local/surgemail/surgemail_start.sh

Upgrading from other Mail Servers and Migration support

SurgeMail has a number of features to make the migration from an existing mail server to your new SurgeMail installation easier. These can be combined to your choice to make your mail migration as painless as possible for both you and your users. However migrating from one mail server to another should always be done carefully and should be fully controlled. Important considerations are factors such as simplicity of upgrade procedure, length of downtime and ability to rollback.

The recommended upgrade method is [migration using POP or IMAP intercept mode](#). This is a simple method to upgrade from any arbitrary mailserver.

Further migration support is described on the [migration page](#).

Upgrading from DMail

SurgeMail will detect an existing DMail configuration and create a SurgeMail configuration based on the `dmail.conf` file. In general this will successfully upgrade an existing DMail installation.

However, SurgeMail is not 100% compatible with DMail and upgrades should be attempted with caution. It is suggested to copy your existing `dmail.conf` file to a test systems and install SurgeMail on that to test that SurgeMail will upgrade your DMail configuration.

The suggested upgrade procedure is to copy your existing `dmail.conf` file to a test systems and install SurgeMail on that to test that SurgeMail will upgrade your DMail configuration. The SurgeMail installation scripts will warn you about any ini settings that it thinks it cannot convert and might be serious. If you use include files in your `dmail.conf` or have scripts that modify `dmail.conf` you should definitely hesitate as SurgeMail uses a completely different ini file format.

Also note that when DMail is upgraded external modules such as authentication modules and their databases will remain in their original location, it is important to check your `dmail.conf` and `surgemail.ini` files before deletion of any files relating to your DMail configuration.

Please read the following [online upgrade notes](#) for any latest information on upgrading DMail.

Uninstalling SurgeMail

Windows

The uninstall is available through the Start Menu - SurgeMail - Uninstall SurgeMail, or through Control Panel - Add / Remove Programs. If for some reason this is not available you can also run "surgemail -uninstall" from the command line in the directory that SurgeMail is installed.

UNIX

Run `./surgemail -uninstall` in the directory in which SurgeMail was installed. (default is `/usr/local/surgemail`)

Note: Have you read [How SurgeMail fits together](#)? *It will help your understanding of SurgeMail features.*

Mailserver Migration

SurgeMail has a number of features to make the migration from an existing mail server to your new SurgeMail installation easier. These can be combined to your choice to make your mail migration as painless as possible for both you and your users. However migrating from one mail server to another should always be done carefully and should be fully controlled. Important considerations are factors such as simplicity of upgrade procedure, length of downtime and ability to rollback.

Unless you have specific requirements, the recommended method for upgrading to SurgeMail is POP / IMAP intercept mode. We aim to make the migration as simple and safe as possible. If you have any suggestions on how we can improve the migration process please let us know:

surgemail-support@netwinsite.com

Note: In all cases we recommend you start by making a **backup copy** of your existing mail server's mailbox files :-)

Migration using POP / IMAP intercept mode (recommended)

This is a simple zero downtime method for migrating all active accounts from any arbitrary mailserver to SurgeMail. This method configures SurgeMail to be the new mailserver for all users. When a user logs into SurgeMail and a local account does not exist, SurgeMail will login to the old server check the account is valid. If the account is valid SurgeMail will create an account and retrieve mail for this account storing it locally. From now on the user will use SurgeMail as their primary mailserver and mail for them will be delivered locally. Mail delivered to SurgeMail for users that have not yet logged in will be forwarded on to your old mail server.

1. Backup your existing mail server's mailbox files :-)
2. Install SurgeMail onto your new server and configure to host your existing domains
3. Setup the options [old_POPhost](#) / [old_IMAPhost](#) and [fallback_relay](#) for the domain you are migrating (on domain administration page). The fallback_relay setting will pass on all mail for accounts that not yet have been created locally to your old server and the old_pophost setting will allow SurgeMail to retrieve account information and email from the old mailserver.
4. Change your DNS and MX records to point to surgemail
5. Whenever a user checks their mail an account with the same username and password will be created in SurgeMail and all outstanding mail will be retrieved.

Detailed examples and documentation can be found [here](#) on using this method. Note! migration cannot be used in conjunction with CRAM-MD5 as that prevents surgemail from storing the old password. CRAM-MD5 is best not used as SSL offers better security.

You can also use this command line tool to import a single account or test importing an account. Examine migration.log if it is not successful:

```
tellmail imap_import newuser@domain oldimaphost olduser oldpass delete|keep create|nocreate
```

e.g. to import an account, without deleting the messages from the old one, use:

```
tellmail imap_import test@xyz.com mailsever.xyz.com test@xyz.com SECRET keep create
```

If you wish to test an account by logging in to it, and then you wish to delete the account and test it again you would need to issue this command so that surgemail will know it needs to retry:

```
tellmail pstat_delete user@domain.name
```

Migration from Non SurgeMail server

SurgeMail has a number of features to make the migration from an existing non SurgeMail mailserver to SurgeMail easier:

- Parallel operation / gradual migration
- Authentication database import
- Delivered Mail conversion
- Complete configuration upgrade (DMail only)

Parallel operation / gradual transfer

SurgeMail has the option "[fallback_relay](#)" (configurable per domain) that allows mail to be delivered to a different host if the user does not exist locally. This can be used whilst testing that a system is operational and if desired for the gradual migration of users from an existing server to the current server. The two ways of configuring this are as follows:

1. Configure SurgeMail to handle the domain served by your existing server. Keep your MX record pointing at your existing server. On the existing server cc (or redirect) mail to your SurgeMail configuration. For the account(s) in question start using SurgeMail to retrieve and send mail. In this configuration rollback is easy by simply switching your mail client back to the original server. Note: If you are sending to servers that do a reverse lookup you will need to get SurgeMail to relay outbound through your existing server.

2. Configure SurgeMail to handle the domain served by your existing server. Switch your MX record to the SurgeMail server. Now if the account in question is defined in the SurgeMail user database, SurgeMail will process mail for this user and the users mail client will need to point to SurgeMail. If the account is not present all mail will be forwarded to your existing mailserver and the users mail client will need to point to your existing mail server.

Note: Again if you are sending to servers that do a reverse lookup you will need to get your existing server to relay outbound mail through the SurgeMail server.

Authentication database import

There probably is no need to is no need to import your existing user database as SurgeMail is very likely to have an authentication module that will use your existing database directly. Having said that many administrators do choose to convert their existing user database to NWAAuth. This is especially so if using system based authentication as NWAAuth does not require a UNIX system account for every user in your authentication database.

If converting your sendmail configuration your existing /etc/passwd + /etc/shadow account database can be imported to NWAAuth format without users having to change their password or the encrypted password being decoded either by manually copying the fields or using the following script [xferusers.pl](#).

Delivered Mail Conversion

SurgeMail uses maildir format to store delivered mail. SurgeMail has the inbuilt ability to convert standard drop files to maildir format. In addition SurgeMail will convert dmail bin files and mbx files to SurgeMail mdir format. The settings that control this process are the surgemail.ini vdomain settings: dmail_drop_path, dmail_bin_path and dmail_hash.

eg: If you were converting a Windows DMail configuration the upgrade settings would be as follows, these are automatically added if you do SurgeMail install based on a dmail.conf.

```
dmail_bin_path "D:\somewhere\binfiles"
dmail_drop_path "D:\somewhere\dropfiles"
dmail_hash "0"
```

Note: Hash level: 0 = no hashing, 1= [dmail_specific](#) hashing, 2=generic double level hashing (mail/f/r/fred), 3= [dmail_specific](#) hashing.

eg So unhashed UNIX Sendmail drop files can be converted as follows:

```
dmail_drop_path "/var/spool/mail"
dmail_hash "0"
```

The default SurgeMail behaviour is to check for mail to convert when a user logs in using POP or IMAP. This means that normally the conversion load is evenly spread over the timeframe that users login. However larger accounts may see a delay in checking their mail while the mail conversion takes place. This can be prevented by doing the conversion of all "outstanding old mail" in batch mode using the following command:

```
tellmail convert_dmail
```

This command requires dmail_bin_path and dmail_drop_path and dmail_hash settings to have been setup correctly prior to running the command.

DROPPFILE format (pine)

For pine format unix drop files (Each folder is contained in a single file with each message starting with the line 'From ...', you can use this command to import mail folders (other than the inbox which the dmail conversion commands will handle) This commands requires SurgeMail 3.7c4 or later.

```
tellmail dropfile_import john@your.domain /home/john/mail
```

Complete Configuration Upgrade

As each mail server configuration tends to be unique and specifically configured we do not support a generic upgrade ability from an arbitrary mail server. We aim to supply the capability to easily migrate the necessary elements of your configuration to surgmail.

Having said that, SurgeMail does have the ability to upgrade an existing DMail configuration. It does this by generating a surgmail.ini configuration based on the dmail.conf configuration. This reuses the authentication database that DMail used and adding the settings described in "Delivered Mail Conversion" above. Then when a user logs in mail existing mail is converted to the correct SurgeMail location.

Moving a SurgeMail configuration

SurgeMail has been designed to make migration between two SurgeMail servers of similar or of different platforms very easy but again there are several different ways to migrate. **We recommend the [Mirroring migration method, see instructions here](#).**

NOTE: If you **upgrade to the latest surgmail version** in the process of transferring to a new server, then you will need to purchase updates (if you don't currently have updates). Alternatively, you can install a matching 'old' version of surgmail on the new system. (Old releases are available [here](#)).

Migration by copying directory tree

The easiest and currently the recommended way of moving SurgeMail between servers of the same platform is by copying the whole SurgeMail directory tree. To do this, install a default configuration on the new server, then copy the entire SurgeMail directory tree (plus /etc/surgemail.ini) to the new server, move your license key from one host to the other using tellmail deactivate / activate, and update your mx record.

"Moving" your SurgeMail key:

1. If you want to temporarily run your two servers side by side, you can run with the normal temporary trial key until you transfer your license.
2. Run "tellmail deactivate N##### myemail@mydomain.com" on the old server to disable your paid key on your old server
3. Run "tellmail activate N##### myemail@mydomain.com" on the new server to enable your paid key on your new server

Migration by copying data files

Alternatively the relevant data files can be copied. This is the recommended way when moving between platforms or you wanting a clean install on the same platform. To do this you will need to install a default configuration on the new server, then copy the following configuration, mail and data files from the old installation to the new installation, move your license key, and update your MX record.

To copy SurgeMail data directories you will need to copy:

1. surgmail.ini (c:\winnt for windows or in /etc for unix) : main SurgeMail configuration
2. surgmail_directory\mydomain.com : delivered mail for each domain.
3. surgmail_directory\webmail_work : WebMail options (and folders if running WebMail in POPmode)
4. surgmail_directory\scripts\webmail.ini : WebMail configuration file
5. surgmail_directory\nwauth.txt + nwauth.add : user database (if using NWAAuth)
6. surgmail_directory*.ini *.dat : variety of configuration files
7. surgmail_directory\work (optional) : undelivered mail queue
8. surgmail_directory\recYYMM (optional) : delivery record
9. surgmail_directory\ssl (optional) : if signed certificates have been added these will need to be copied
10. surgmail_directory\bull and dlist (optional) : configuration of bulletins and mailing lists

On UNIX be sure to correct the file ownerships to 'mail' if you copied the files as root! (chown -R mail /usr/local/surgemail)

Zero downtime migration using Mirroring (RECOMMENDED)

[See this page for detailed instructions](#)

In both the above two cases neither the old or new SurgeMail installations should not be running whilst copying files. If this is not acceptable in terms of interruption of service you can use migration using pop intercept mode or migration using mirroring. The optimal procedure is still being finalised for zero downtime upgrades so please test your chosen migration path on a non live

system first and again suggestions for improvement are welcome.

The mirroring feature can be used to migrate user accounts and mail from one server to another using zero downtime. To do this you need to Setup your new server in the same configuration as your old server by copying the configuration files and install a temporary second license key. Enable mirroring on both systems. Add an MX record with higher priority for your new server. At this stage both servers will be live and mail can be delivered to either server and it will be mirrored to the other server. When you are confident the new server is working as expected you remove the mx record for the old system, wait for any remaining mail to be mirrored to the new server and take the old server offline.

Several things should be carefully noted. Firstly mirroring only duplicates delivered mail, more specifically files that get mirrored are the files stored in the surgemail/mydomain.com maildir directories for each domain. This means that certain settings or files will not get mirrored as follows:

1. If you are using WebMail in POP mode mail stored in WebMail folders will not be synchronised. If you are using WebMail in IMAP mode (since recently the default) mail in WebMail folders will be copied.
2. WebMail user configuration settings, DList information will not be copied.

Miscellaneous Mail Conversion / Import / Export

Moving mail in WebMail folders to surgemail IMAP folders

When running WebMail in POP mode the mail stored in folders is stored in the WebMail work area. When running WebMail in IMAP mode this mail is stored in the SurgeMail folders and is accessible using other IMAP clients. To convert this mail do the following:

1. You need WebMmail version 3.0r or later
2. Ensure you have a manager password configured in your webmail.ini file "managers_password mypassword"
3. Run "webmail.exe -manager", do option "m" and enter "*" for converting all users.

Importing old mail from other Maildir implementations

Maildir implementations vary slightly in their implementation. SurgeMail is able to import other Maildir formats by running the command "tellmail maildir_convert surgemail_domain.com source_directory" where source directory is a fully specified path with \$HASH1 / \$HASH2 and \$USER, where expansion of directories is required.

```
tellmail maildir_convert mydomain.com '/home/$USER/Mail'
tellmail maildir_convert mydomain.com '/home/$HASH2/$USER/Mail'
tellmail maildir_convert mydomain.com '/var/spool/mail/$HASH1/$USER/Mail'
```

Where

\$HASH1 = /fr (for fred)

\$HASH2 = /f/r (for fred)

\$USER = wild card scan of usernames

eg: Converting all mail stored in the format /var/spool/mail/user1/Mail to /usr/local/surgemail/mydomain.com/bb/wj/user1/

```
tellmail maildir_convert '/var/spool/mail/$USER/Mail'
```

Virtual User Table - Translating username to username@domain.name

Using 5.1 or later you can create a file called 'virtusertable.dat' to translate user names given without a username during pop/imap login into a user and domain name pair, this should only be used when converting users from an old sendmail system where you need to translate the users to multiple different domains, e.g.

fred -> fred@domain1.com

bob -> bob@domain2.com

joe -> joe@domain3.com

The file format is as follows:

```
fred fred@domain1.com
bob bob@domain2.com
joe joe@domain3.com
```


Post Installation Instructions

1. Check your free trial license key (optional)

- If all goes well at install time your 30 day trial license will already be installed and SurgeMail will be fully functional.
- You can check this using the Web Admin interface (<https://your.server.com:7025>)
- Select the Register page (click "Register" link on left hand navigation pane)

2. Creating a user

- To create users, select Accounts from the left hand navigation pane.
- Enter a username and password and click on Create.

3. Ready to send mail :-)

- You can use the installed WebMail client to send and check your mail straight away. Select the "Web Email Client" link on the "Welcome To Surgemail" page.
- Alternatively you can connect a mail client to your server for POP or IMAP access to the account.

Further configuration

You will typically want to do quite a lot of further configuration to tailor and secure your system. These include configuration of your multiple virtual domains, possibly limiting administration to certain IP addresses, maybe tailoring the customisable templates eg: </surgemail/web/index.htm>, and converting the Web HTTP Port to 80 so that users can use your mail system by just typing in your domain name.

You probably will need to:

Setup your DNS/MX Records

If you don't already have a DNS/MX record for your system you will need one before other people can send email messages to your new domain name from the internet. Ask the person who runs your DNS server to add a DNS and MX entry.

Select an authentication Method

By default, SurgeMail will use 'NWAAuth', a simple internal user database, suitable for any number of users. However, you may wish to select a different one, eg:

- UNIX /etc/passwd based
- Windows NT/2000 user database
- MySQL, LDAP, Radius etc...

To do this select Modules from the navigation pane.

Setup your DNS host

SurgeMail needs a DNS server to send to remote email addresses. SurgeMail will typically use the default operating system settings but if outbound mail is not getting sent you may need to specify this manually on the Global settings page. You would use the IP address of whatever upstream DNS server you would normally use eg: your ISP DNS server.

SurgeWall

SurgeMail can act as a gateway / mail filter for a pre-existing mail server. This allows you to give users of that server access to SurgeMails spam, virus and friends features, which are configurable through the web interface. For more information about configuring SurgeWall see [surgewall configuration](#) and [surgewall settings](#).

Getting Help

Still stuck? If so, search the manual and examine the log file. If that fails email our support staff at: support-surgemail@netwinsite.com ideally send us your config file and log file eg: \winnt\surgemail.ini and \surgemail\mail.log on windows (or /etc/surgemail.ini and /usr/local/surgemail/mail.log on unix) and a good description of what's going wrong.

Note: If we tell you about a setting (or you hear about a particular setting eg g_dns_host) you can easily find this in the web interface by entering this in the 'find config setting' search field in the top left corner and pressing enter.

Note: Have you read [How SurgeMail fits together?](#) *It will help your understanding of SurgeMail features.*

FAQ - Frequently Asked Questions

- [How do I restart the server?](#)
- [How do I get to the web manager?](#)
- [How do I set the web manager password?](#)
- [How do I enter my registration key?](#)
- [Why is the manager sluggish on Netscape & Win 2000?](#)
- [How do I stop swatch \(SurgeMail Monitor\)?](#)
- [Concurrent user limits.](#)
- [Slow performance / virus scanner installed](#)
- [How do I tailor the quota message](#)
- [How do I use SurgeMail with Pine / Mutt](#)
- [Where are the SurgeMail configuration files](#)
- [Why doesn't my WebMail auto-login work?](#)
- [What does "Failed to create Pass ID" mean?](#)
- [How do I get WebMail to allow me to move files back to the inbox?](#)
- [Why do I get the error "Failed to detect host" in WebMail?](#)
- [Where can I download / get a printed version of the manual?](#)
- [How can I use "sendmail" style command line syntax to send mail?](#)
- [How do I fix "DNS lookup failed" errors?](#)
- [How do I fix DNS "lookup_name" lockups?](#)
- [How do I add users on the command line?](#)
- [Is there a printed version of the documentation?](#)
- [What ports do I need to open on my firewall?](#)
- [What does the SurgeMail version numbering mean?](#)
- [WebMail fails to connect why?](#)
- [Can I run WebMail on a seperate machine?](#)
- [How do I \(manually\) install/uninstall Avast antivirus?](#)
- [I have more than one IP but only want SurgeMail to listen on one, how?](#)
- [How do I check my virus scanner is running?](#)
- [How do I make SurgeMail bind to one ip for outgoing messages?](#)
- [How do I move SurgeMail to a new machine?](#)
- [How do I backup SurgeMail?](#)
- [How do I send my mail through my ISPs server?](#)
- [Can I use include files in surgemail.ini ?](#)
- [AOL or some other domain won't accept mail from me because I'm a dsl/cable user, what can I do?](#)
- [How can I change a domain name in SurgeMail?](#)
- [How do I run SurgeMail and IIS SMTP virtual server on the same Windows server?](#)
- [Help I cannot delete a message in webmail - turn off your virus scanner!](#)
- [I can't receive mail](#)
- [I can receive email but not send it](#)
- [I am getting a DNS lookup failure, what can I do?](#)
- [I am running Apache/IIS hosting a website. Can I run SurgeMail with webmail on the same machine?](#)
- [How do I rename a domain?](#)
- [How can I fix this error 399 TCP Read failed?](#)
- [Duplicate messages sending or receiving](#)
- [How do I get to the command prompt in Windows?](#)
- [How do I whitelist a domain?](#)
- [Sendmail - My forms can no longer send mail](#)
- [Why do some messages appear multiple times in a users inbox?](#)
- [AOL have blocked my server, what can I do?](#)
- [SMTP session numbers growing, g thread max, g smtp_max, g pop_max limits](#)
- [NoSubmit errors on incoming email](#)
- [How do I move spam to a spam folder for all users](#)
- [How do I setup SurgeMail to be a backup server for a domain?](#)
- [In what order do the various parts of SurgeMail run? \(g virus cmd, g virus filter etc etc\)](#)
- [To convert a PFX file to a PEM file, follow these steps on a Windows machine](#)
- [SurgeMail/SurgeNews crash every minute on 64bit linux](#)
- [libgcc s.so.1 must be installed for pthread_cancel to work](#)

- [Setting up ATRN service for a client](#)
- [How can I stop local accounts being compromised and used for spamming?](#)
- [How do I restrict who some accounts can mail?](#)
- [Blacklisting due to responders friends bouncing etc ironport filtering or spamcop](#)
- [554 Failure tcp read dot](#)
- [Email fails with error "No DATA command sent-rset"](#)

How do I restart the server?

On NT, type in: **net start surgemail**

On 95/98, type in: **/surgemail/surgemail**

On Unix, type in: **/usr/local/surgemail/surgemail_start.sh**

To start it remotely use SurgeMail monitor on <http://your.mail.server:7027>

How do I get to the web manager?

Use your web browser and type in a link like this: <https://your.mail.server:7025> If you are on the machine itself this link should work <https://127.0.0.1:7025>

How do I set the web manager username and password?

*nix systems you need to go to the shell and type

```
cd /usr/local/surgemail
```

```
./surgemail -password
```

Windows systems you need to go to the [command prompt](#)

```
cd \surgemail
```

```
surgemail -password
```

Can I use include files in surgemail.ini

Not normally no, the reason for this restriction is that the web interface allows you to modify config settings in a nice and safe manner, as a result the ini file needs to be re-written, and if it was composed of include files it would be difficult, dangerous and very nearly impossible to write it out correctly after a change.

However, you can do it if you want, but if you put an 'include full_path/file.ini' directive in surgemail.ini then any call to save the ini file will silently fail (so your web admin will just seem to work but won't really). Also note, if you create an ini file with 10,000 include files in it, and it takes 3 minutes to reload it, we will not have any sympathy for you (we've seen this done before really!)

How do I enter my registration key?

In the web manager click on the 'Register' link on the navigation pane.

Alternatively run tellmail activate on the command line with your registration details:

```
tellmail activate N123 myemail@mydomain.com
```

Why is the manager sluggish on Netscape & Win 2000?

This is a bug in Netscape. It steals all the CPU while waiting for a web page to arrive but since the server is on the same system that means it responds slowly. You can fix it in task manager: set the priority for Netscape down to 'below normal' and suddenly it will work faster! Or upgrade to a fixed version of Netscape.

How do I stop swatch (SurgeMail Monitor)?

It should never be necessary to stop swatch manually as swatch is designed to keep running so that SurgeMail can be restarted using the web interface. But, if for any reason the SurgeMail monitor process needs to be manually stopped create the mon.exit file in the SurgeMail directory and swatch will shutdown.

Are there concurrent user limits?

The number of concurrent users is operating system dependent and basically a matter of how many threads and file handles the operating system supports. Here are the approximate figures-

Operating system	Concurrent mail sessions (these are not hard limits)
Windows NT	1,500
Linux	500-1000 on early versions more on recent kernalns that do not have handle or thread limits
Solaris 7	500
Solaris 8	2,000

How do I tailor the quota message?

In the SurgeMail directory create a file called quota.eml, something like this (this requires SurgeMail 1.3m or later):

```
Subject: A new quota message ||domain||
reason: ||reason||
max=||max|| used=||used|| size=||size||
This is the quota
message for domain: ||domain||
ends here.
```

Which will look something like this for the user:

```
Subject: A new quota message xxx.yyy.com
reason: Quota exceeded 250000>200000
max=200000 used=190000 size=60000
This is the quota
message for domain: xxx.yyy.com
ends here.
```

How do I use SurgeMail with Pine / Mutt to read my mail?

SurgeMail uses mdir format to store mail which cannot be read by mail clients that read the mail drop file directly. The "Deliver" mail delivery robot can be used to deliver mail to a drop file:

Deliver is available from sourceforge.net and can be configured in SurgeMail using a mail redirection rule in surgemail.ini as per:

```
g_redirect was="marijn@mydomain.com" to="|./deliver -b /var/surgemail/mydomain.com/hd/fg/marijn/dropfile"
```

Where are the SurgeMail configuration files?

SurgeMails main configuration file is surgemail.ini which is store in /etc on UNIX systems and your Windows directory on Windows systems (eg c:\winnt). This file can be edited by hand after which a "**tellmail reload**" would need to be issued or edited via the web interface. Backups of this file are stored in the SurgeMail directory as ini_YYMMDD.rec.

WebMail has a separate configuration file stored in surgemail/scripts/webmail.ini.

How do I get WebMail to allow me to move files back to the inbox?

If WebMail is using IMAP to talk to SurgeMail this can be enabled using the following setting in webmail.ini. (This is now enabled by default but used to be disabled by default)

```
enable_inbox_transfer "true"
```

Why do I get the error "Failed to detect host" in WebMail?

This just means WebMail could not talk to SurgeMail. There could be several reasons. The most likely reason is that SurgeMail does not have the correct settings in the event that your domain name is different from your hostname eg mydomain.com vs mail.mydomain.com [see for more detail](#)

Check your webmail.ini file, surgemail/web_work/surgehost.ini and surgemail.ini for possible misconfiguration of individual domains.

Where can I download / get a printed version of the manual?

The online help is the primary documentation this is distributed with every SurgeMail download and the latest version is available [online](#). An automatically generated [pdf version](#) is also available.

How do I install additional WebMail templates?

Different WebMail templates may be installed. The Surgemail + Webmail distributions come with three template sets by default (Panel, Surge and Smooth). Several additional template sets are available but most of these have a rather out of date "look and feel" to them and or do not supply all the functionality now supplied by webmail and surgemail (these include marble, iconic, vanilla).

Several examples of the flexibility of the webmail look and feels can be found on the the following pages:

<http://netwinsite.com/surgemail/templates.htm>

<http://netwinsite.com/webmail/gallery/index.htm>

All that is required to install a template set is to add the actual template files to the directory surgemail/webmail/templatename, the images to surgemail/www/nwimg/mail/template name and add one line to webmail.ini defining it. eg:

```
tpl_set 2 E:\surgemail\webmail\marble /nwimg/mail/marble Marble Set (Marble)
```

How can I use "sendmail" style command line syntax to send mail?

SurgeMail installs a sendmail stub. This will allow your PHP scripts and the like to continue sending mail using the same syntax they have always done. You will need to ensure SurgeMail is allowing relaying for your local IP. If it is not working pass the stub the "-debug" parameter it should create a sendmail.debug file that will give you information as to why it is not working.

How do I fix "DNS lookup failed" errors?

This means that DNS resolution of an address failed and can be for one of several reasons:

- 1) Wrong server being used
- 2) Server is not responding or firewall / router is blocking TCP port 53

SurgeMail will attempt to use the DNS settings of your operating system for its name resolutions. If this is not working for some reason you can manually force SurgeMail to use particular dns servers using the setting g_dns_host setting. eg. where the IP numbers are the ip addresses you wish to force SurgeMail to use.

```
g_dns_host "1.2.3.4,2.3.4.5"
```

note: You must restart SurgeMail when changing g_dns_host

SurgeMail provides status information on the DNS servers that it uses in on the status page on the web interface.

If this still fails it may be that the DNS server is faulty and is not responding or that a firewall or gateway is blocking TCP port 53 access. (some OS services only require UDP access which is why your firewall might be blocking TCP traffic on port 53) To test this telnet to your DNS server as per "telnet your.dns.server.ip 53". If this does not connect this is the problem. If this does connect then your DNS server is working fine.

How do I fix DNS "lookup_name" lockups?

Set the setting g_dns_paranoid to false i.e.

```
g_dns_paranoid "false"
```

And restart SurgeMail. If the problem persists contact surgemail-support@netwinsite.com.

How do I add users on the command line?

Users can be added on the command line as follows (You need to run the path etc for your authentication module as specified in surgemail.ini)

```
./nauth -path . -set username@domain password
```

Alternatively for a more efficient process create a text file of nauth commands and pipe it to nauth as follows:

```
>>Start of file nauth.in<<
```

```
set user1@domain.com password
```

```
set user2@domain.com password2
```

```
set user3@domain2.com password3
```

```
>>End of file<<
```

```
./nauth -path . < nauth.in
```

or for list of nauth command line commands:

```
./nauth -help
```

In new versions built later than the 25th of April 2004 you can now use a tellmail command

```
tellmail add_user <user@domain> <password>
```

This automatically uses the correct authent module etc

Is there a printed version of the documentation?

Yes, a pdf version of the online help can be downloaded from:

<http://netwinsite.com/ftp/surgemail/doc/surgemail.pdf>

What ports do I need to open on my firewall?

This depends on the services you wish to offer, but in principle the main ports you will need open to TCP traffic are:

53 DNS lookup for outgoing mail

110 POP3 services (Also used for mirroring)

143 IMAP services

25 SMTP services

80 (or 7080 if port 80 is already in use) Webmail HTTP access

7025 Administration HTTPS access

SurgeMail also uses the following

995 Secure POP3 services

993 Secure IMAP services

465 Secure SMTP services

7110 SurgePlus

7443 Secure Webmail HTTPS access

7026 Administration HTTP access

7027 Monitor HTTP access

What does the SurgeMail version numbering mean?

SurgeMail version numbering is setup as follows <Number>.<number><letter>[optional number] - <build number> eg 1.5a - 12

- The first number is a major release number version expected to change approx once per year.
- The second number is a release number which will include new features and is expected to change once a month.
- The letter is a sub version number which gets modified each time a new version is uploaded to netwinsite either as a beta or as a specials build. This will probably change almost on a daily basis and will include both new features and bug fixes. eg 1.5a -> 1.5b
- If the production builds need to be patched for a specific bug fix only, the optional number will be incremented eg 1.5a2
- Whenever a new build is supplied to a customer the <build number> will be updated (as of SurgeMail 1.8e)

[Updates.htm](#) documents changes since the last production release whenever a new version is released as a likely production release candidate.

Prior to version 1.5a this was defined slightly differently.

WebMail fails to connect - Failed to Auto-Detect POP or IMAP at

In surgemail/webmail.ini ensure the settings for IMAPhost and SMTPhost point to your actual server and not a domain that resolves to some other system, or doesn't resolve at all eg:


```
smtp host localhost
imap host localhost
```

Can I run WebMail on a separate machine?

Yes. In some cases it might be beneficial to run the WebMail CGI on a different machine, to do this, simply install WebMail on the other machine see the [WebMail documentation](#) on how this is done. Then in addition to the normal configuration requirements, eg: pophost smtp host etc you need to configure these WebMail settings:

```
use_id_autologin true
```

```
friends_only true
```

```
autorespond true
```

```
netwin_autologin_id 0 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_load_pass&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 1 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_load_fcommon&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 2 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_load_fwd&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 4 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_spam_load&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 5 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_load_centipaid&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 6 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_sms_load&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

```
netwin_autologin_id 7 https://surge mail.server.com:7025/cgi/user.cgi /var/spool/webmail lcmd=user_listmb&vhost=|vhost|&webmail=true&
```

```
bgcolor=|href_text|cust_panel_bgcolor|&tdcolor=|href_text|#eeeeff|&thcolor=|href_text|#D6D6CE|&border=0&background_image=|href_text|cust_panel_ba
```

And these surge mail.ini settings:

```
g_autologin_pop "TRUE"
```

```
g_webmail_url "http://other.server.com/scripts/webmail.exe"
```

In addition for every domain you add to SurgeMail you will now manually need to update webmail.ini with the domain details, see the [WebMail documentation](#) on "Virtual Hosts" for how this is done.

How do I (manually) install / uninstall Avast Antivirus

The avast installer is integrated to SurgeMail web admin interface - just press the install and uninstall button on the globals page.

There should not be the need to manually install Avast but if necessary this can be done by: downloading the installation package from [ftp://netwinsite.com/pub/surge mail/util/avastoem.exe](http://netwinsite.com/pub/surge mail/util/avastoem.exe) running the command line:

```
avastoem.exe /oem "SurgeMail"
```

and making sure it is installed into the SurgeMail\Avast directory.

Again there should not be the need but to manually uninstall just delete all the files and subdirectories in the \surge mail\avast directory other than surge mail\setup\setupif.dll which is required to install again via surge mail web admin. In addition you need to delete the registry key :HKEY_LOCAL_MACHINE\SOFTWARE\WIL Software\Avast\SurgeMail and all entries within it.

I have more than one IP but only want SurgeMail to listen on one, how?

`g_smtp_port <ip:port>`

This allows SurgeMail to listen on a specified port and IP, you can add multiple IPs if you wish to listen on more than one and multiple ports also.

eg:

`g_smtp_port "1.1.1.1:25, 2.2.2.2:1025"`

How do I check my virus scanner is running ?

You can check the status page and check how many viruses have been caught. You can also send a test virus through which can be got from www.eicar.org, and then of course there are the logs you can check.

How do I make SurgeMail bind to one ip for outgoing messages?

In `surgemail.ini` add the following setting then restart.

`g_bind_out "x.x.x.x"`

How do I move SurgeMail to a new machine?

There are two ways of doing this, one is basically copying all the files to the new machine. The second is by setting up a mirror and letting SurgeMail mirror itself over to the second machine.

[See this page for the mirroring method \(recommended\)](#)

Or the manual method to move SurgeMail

1. Install SurgeMail on new machine
2. Setup anything you tailored on the original system (e.g. authent modules)
3. Stop SurgeMail on new machine
4. tellmail deactivate on old machine
5. Stop SurgeMail on old machine
6. Copy `surgemail.ini` from old machine to new machine
7. Copy the SurgeMail directory, the mail directory and the database over to new machine.
Check `surgemail.ini` for the paths to copy for the mail directory (`g_mailbox`)
The default database is `nwauth` which will be stored in the `surgemail` directory but consists of the files (`nwauth.add`, `nwauth.txt`)
8. Check `surgemail.ini` and check everything is located correctly. Change all the paths if necessary
9. Chown -R mail files (if on UNIX) for all mail folders etc...
10. Start `surgemail` on new machine, check logging in etc
11. tellmail activate on new machine
12. If you have moved from one OS to a different OS then you should run the installer on the new server again so that it places the correct binaries on the new system.

How do I backup SurgeMail?

- Backup `surgemail.ini` which is found in the windows directory or `/etc` if on UNIX
- To backup the mail you need to backup the directory that stores all the mail. You will need to check this location in `surgemail.ini` look for `"g_mailbox_path"` and that will give you the directory to backup.
- To backup the user accounts, if you are using `NWAuth` for your user database (which is the default) then you should backup all the `nwauth*` files in the SurgeMail directory.
- Finally you can backup the SurgeMail directory which contains the programs, the temporary work area for queued messages, and the templates etc etc.

On Windows you can just use something like winzip to copy everything, on UNIX based platforms you can use tar and gzip or whatever you prefer.

To restore a backup

1. Install SurgeMail on the new box
2. Shutdown SurgeMail on both machines
3. Copy your backup over to the new machine and `untar/unzip` (If you are on linux using tar is a good idea as

this will preserve file permissions)

4. Check the file permissions if you are linux (ls -l) they should be owned by mail
5. Copy surgemail.ini over to the new machine and check the paths in it to make sure they are correct.
6. Restart SurgeMail

If you are moving SurgeMail to a new machine you can check this guide

http://www.netwinsite.com/surgemail/help/faq.htm#moving_surgemail

How do I send my mail through my ISPs server or some other mail server?

If you need to send your mail via another SMTP server then you can use the gateway setting. This setting lets you choose which domains to send to a server so you can send one domain to one server and another domain to another server or you can send all domains to one server. This is useful if your ISP won't allow you to connect to port 25 on remote machines or if you are on cable/DSL and domains like AOL won't accept mail from you because of this so instead you can send all your mail through your ISP's mail server.

Example 1: Sending all mail through a different server

```
g_gateway domain="" to="ip of server to use" relay="false"
```

If you need to authenticate on the server you are going to use you can do this

```
g_gateway domain="" to="ip of server to use" relay="false" user="user to auth with" pass="password of user"
```

Example 2: Sending mail going to AOL via a different server

```
g_gateway domain="aol.com" to="ip of server to use" relay="false"
```

You can add SMTP AUTHentication like in example 1.

You can find more information on using the gateway setting here [g_gateway](#)

How can I change a domain name in SurgeMail?

1. Stop Surgemail
2. Make a backup of nwauth.*
3. Run **./nwauth -rename old.domain.name new.domain.name**
4. Edit surgemail.ini and again change domain names. **Note do not change mailbox_path setting** (or if you do rename the physical directory to match)
5. Change the name of the mailbox directory [default will be old domain name] if modified in ini file above.
6. Edit webmail.ini and change old to new domain name. Check IMAP and SMTP server names if appropriate.
run `./surgemail/scripts/webmail.cgi -manager`
7. Select s option and specify old/new domain names for converting across users.
8. Restart Surgemail and using the web interface, correct any aliases and/or virtual domains which may have redundant new domain names.

Slow performance, virus scanner installed?

If you have a virus scanner like Nortons installed with Auto Protect enabled, you should disable the auto protect feature which will seriously kill performance of the mail server (e.g. it may run 100 times slower). Also if you have your virus scanner enabled for incoming/outgoing mail you should disable that as well as it could easily break the mail server protocol in unexpected ways.

SurgeMail has a virus scanner option 'avast' which should be used for scanning for viruses in mail messages, it is much much more efficient as it is properly integrated.

I cannot delete a file in webmail (turn off your virus scanner!)

If you have a virus scanner like Nortons installed with Auto Protect enabled, it may lock access to webmail files, this will prevent webmail from deleting the message file, this then upsets the user :-). So let me repeat, don't run a virus scanner on your mail server, instead use a scanner inside SurgeMail. (e.g. avast)

How do I run SurgeMail and IIS SMTP virtual server on the same Windows server?

Applies to Windows 2000

PROBLEM: You can not have SurgeMail and IIS SMTP Virtual Server listening on the same TCP port 25. The IIS SMTP service listens to port 25 on all unassigned IP addresses, even though you specify a specific IP address for

the SurgeMail server. You need to disable the MS IIS socket pooling feature (DisableSocketPooling). This property is not exposed in ADSI for SMTP.

This works for IIS also, just substitute Smtplib with the W3SVC

SOLUTION:

1. Assign a unique IP for SurgeMail For example if your server's IP is 123.123.123.30 , assign 123.123.123.31 to be used for the DNS resolution that will point to the URL that resolves to SurgeMail on your Windows server, e.g. mail2.yourdomain.com.
2. Download and install the MetaEdit2.2 utility for IIS. – see article <http://support.microsoft.com/kb/232068/EN-US/>
3. Stop the default SMTP Virtual Server in IIS, if you haven't already to have SurgeMail SMTP work on port 25.
4. Create a backup of your IIS metabase by using IIS Manager, right-click on your server and select Backup/Restore Configuration and then create a backup file in case the edit fails.
5. Run the MetaEdit2.2 utility and follow the instructions at <http://support.microsoft.com/default.aspx?scid=kb:en-us:281760>
6. Restart the IIS default SMTP Virtual Server and it should take off fine. We now have SurgeMail and IIS SMTP virtual server listening on port 25 with two unique IP's!

I can't receive mail

Usually this is one of the following

- Firewall not letting traffic in on port 25
- Router not configured to let traffic in on port 25
- ISP blocking incoming traffic on port 25
- DNS records not setup correctly.

So first check that you can connect locally to the server. At the command prompt on the server type

```
telnet localhost 25
```

You should receive a welcome message like this

```
220 mydomain.com SurgeSMTP (Version 3.1b-1) http://surgeemail.com
```

If you get unable to connect then it's probably due to a firewall running on that machine that is stopping surgeemail.

If you are running SurgeMail on a Windows operating system you can restart surgeemail and then check mail.log and check it says "03 10:32:22.89:Info:2156: Listening on (all interfaces:25)"

You can use this page to help test sending mail to your server.

<http://email-test.com>

I am getting a DNS lookup failure, what can I do?

You need to check that you have the port open for SurgeMail to the DNS.

Port 53 TCP and UDP

You can test the DNS is working and SurgeMail has access to it by going to the shell or command prompt and typing

```
nslookup
server= <ip of DNS>
set type=MX
netwinsite.com
```

Don't type the angle brackets :).

You should then get a response back looking like this:

```
netwinsite.com MX preference = 10, mail exchanger = mail.netwinsite.com
```

You can then type exit to exit the nslookup program.

Once you have tested that it works you can make sure SurgeMail is using this DNS.

Login to the SurgeMail webadmin and in the setting search box type
g_dns_host
then edit that setting and put the ip of your DNS in there.
Then click save, then completely stop surgemail and restart it.

You should now be able to send emails without dns problems, there will be times when you will get some dns lookup failures of course.

I am running Apache/IIS hosting a website, Can I run SurgeMail with webmail on the same machine?

Yes you can!, there are several ways in fact.

1. If you have a spare ip on the machine you can make webmail bind to that IP only and then configure Apache/IIS to only bind to the other IP. This means you can have webmail running on port 80 (normal web port) and it won't interfere with Apache/IIS.
2. You can setup webmail so it runs directly under Apache or IIS, you will need to download webmail separately from netwinsite and run the webmail installer and then turn off the webmail port in surgemail as you won't be using that. You would then setup a virtual domain in apache or IIS for the webmail.
3. You can setup a virtual domain in IIS/Apache and then redirect requests to that domain to port 7080 where SurgeMail's web server will then take care of things, this is operating Apache in a proxy mode but with IIS you are just redirecting the browser, unfortunately IIS as far as I know does not support proxy. This is the easiest thing to do and best thing to do if you don't have a spare IP.

If anyone knows of a free proxy support module for IIS let us know. There is one that could be tried <http://www.isapirewrite.com/> but this is commercial, however they do have a trial period I believe.

Options 1 & 3 are the recommended ones and easiest.

Here is exactly how you would do option 3

Apache:

In this example we will use the domain "test.com" test.com is already running a website and we want to add a subdomain webmail.test.com which will go directly to webmail.

So you will need to edit httpd.conf which is found commonly in /etc/httpd/conf

If you skip to the end you will find the virtual domain setup, basically you would have something like this

```
NameVirtualHost *:80
```

```
<VirtualHost *:80>
```

```
ServerName www.test.com
```

```
DocumentRoot /var/www/html
```

```
</VirtualHost>
```

```
<VirtualHost *:80>
```

```
ServerName webmail.test.com
```

```
ProxyPass / http://127.0.0.1:7080/
```

```
ProxyPassReverse / http://127.0.0.1:7080/
```

```
</VirtualHost>
```

That's it, the first virtualhost block is your default domain and should match your DocumentRoot setting in httpd.conf.

The second block is where we setup webmail.test.com and then use the proxy module commands to forward requests to that domain onto the SurgeMail webserver which by default listens on port 7080

You can find full documentation on virtual domains for Apache here. <http://httpd.apache.org/docs-2.0/vhosts/>

IIS 5/6

This is very easy to do.

1. Load up the IIS Manager.
2. Right click on "websites" then click select New then select website which should open the wizard
3. Enter a description (webmail for test.com)
4. The next screen it asks for the ip and the port, just leave these as they are as we are using named based virtual domains, so in the "Host Header" box type "webmail.test.com" (without the quotes. and click next.
5. It then asks for the path for this virtual domain, just make a directory in your document directory for this domain (like... c:\inetpub\www\webmail) and then select that and then next
6. Just give it read permissions
7. After you click next it should finish the wizard, now we need to tell it to redirect everything to port 7080 for this domain
8. Right click on webmail in the list and then click properties
9. Click on the home directory tab
10. Select "a redirection to url" and then enter http://webmail.test.com:7080 and then click ok

Now when you browse to http://webmail.test.com you should go direct to the webmail pages

If you have any problems please email us at surgemail-support@netwinsite.com

How do I rename a domain?

This is a little bit tricky currently, we will look at making this a lot easier in future versions.

First you should stop SurgeMail.

Then edit surgemail.ini find the domain (in this example test.com)

```
vdomain address="" name="test.com"
mailbox_path "C:\surgemail\mbox\test.com\"
```

You need to change these two settings to the new domain (test2.com)

```
vdomain address="" name="test2.com"
mailbox_path "C:\surgemail\mbox\test2.com\"
```

You then need to copy/move the old mail directory to the new location.

C:\surgemail\mbox\test.com\ to C:\surgemail\mbox\test2.com\

You then need to change all the users in the database as these are all stored as user@test.com so you now need to change them to user@test2.com. If you are using nauth, you can use a text editor to do this, just do a search and replace changing the domain part of the username to the new domain.

Now you have done SurgeMail, then you just need to deal with WebMail.

How can I fix this error 399 TCP Read failed?

1. Upgrade surgemail
2. Check and remove any virus scanner installed on your mail server
3. Check your ip address to see if it's listed in any black listing RBL service
4. Try connecting to the destination host from your mail server, and from another system to see if it works or not.
telnet <ip of destination server> 25

Duplicate messages sending or receiving

There are many situations where duplicate messages can occur. Specifically when one server has sent a message but before the receiving server says 'I've got it' a bunch of tests are performed, if these take too long the connection may timeout. In this situation the sending system will resend, but the receiving system believes it has the message and says "I've got it". Big delays like this should not occur normally with surgemail so would be a sign of something

There are many situations where duplicate messages can occur. Specifically when one server has sent a message but before the receiving server says 'I've got it' a bunch of tests are performed, if these take too long the connection may timeout. In this situation the sending system will resend, but the receiving system believes it has the message and says "I've got it". Big delays like this should not occur normally with surgemail so would be a sign

of something wrong.

Also mail clients set to 'leave messages on the server' can get confused in some instances and refetch messages, this also shouldn't happen though. Examine the msg*.rec delivery logs to determine if a duplicate was delivered twice or just 'read' twice by the client.

Here are some general things to check:

- Check if you have a virus scanner on your system or client, if so remove it and see if that fixes it, virus scanners regularly break the smtp protocol :-)
- If your dns is sluggish surbl may be taking too long: g_surbl name="multi.surbl.org" stamp="sc.surbl.org,ws.surbl.org,phishing,ob.surbl.org,ab.surbl.org,jp"
- Check the following settings which are all potentials for 'delays' which might cause this problem. g_badfrom... g_mx_verify...
- Capture the thread in mail.log of an incoming message that takes a long time, then scan down the time stamps to find the 'biggest' gap in time, then you should see the cause.

Here are some links with things to check and notes on outlook issues:

- <http://support.microsoft.com/default.aspx?scid=kb;en-us;292249>
- <http://support.microsoft.com/default.aspx?scid=kb;en-us;317945>
- <http://www.its.caltech.edu/~halweb/pc/software/outlookxp-duplicate.txt>

How do I get to the command prompt in Windows?

1. Left click on the start button
2. Left click on Program files or All Programs
3. Left click on Accessories
4. Left click on command prompt

How do I whitelist a domain?

There are several steps involved as there are various whitelists, for RBL's , ASPAM etc. RBL's & ASPAM

[g_orbs_late](#) "true" (allows RBL based exceptions based on rcpt and from address)
[g_spf_skip_to](#) "*"@domain" (applies to RBL's and ASPAM)
[g_spf_skip_from](#) "*"@domain" (applies to RBL's and ASPAM)
[g_smite_skip](#) "*"@domain.com" (applies to smite scoring and thus friends - this is the source domain)
[g_smite_skip_to](#) "*"@domain.com" (applies to smite score & friends - this is the destination domain)

If you know the IP's of the domain you want to whitelist you can also whitelist based on them.

[g_spf_skip](#) "ip" - skips spf checks for emails from this ip
[g_orbs_exception](#) "ip" - skips RBL checks for emails from this ip
[g_mfilter_skip](#) "ip" - skips mfilter processing for emails from this ip
[g_spam_allow](#) "ip" - skips spam throttle limits, ideal for the ip address of a mailing list server.

Sendmail - My forms can no longer send mail

SurgeMail replaces the sendmail binary with a sendmail stub, this basically pretends to be sendmail and redirects everything to SurgeMail. Your programs should not have any problems but sometimes there are.

- Create a file called sendmail_surge.ini in /etc on *nix or the windows directory on Windows.
- in this file add the following settings
 host 127.0.0.1
 debug true

Then try sending a message with the sendmail binary

/usr/sbin/sendmail -debug

From: yourusername@yourdomain
To: user@whateverdomain
Subject: test

This is a test

.

You can then view `sendmail.debug` to check what has happened. If you still have problems please send us the `sendmail.debug` log and also the output from a `/usr/sbin/sendmail -version` (surgemail-support@netwinsite.com)

Why do some messages appear multiple times in a users inbox?

There are several ways this can occur. Basically there is a known 'issue' with smtp where a timeout/failure during the 'data' stage after the 'dot' is sent by the sender, can result in the receiving system thinking it's got the message while the sending system thinks it failed to send it so retries.

Usually this occurs due to something slightly odd going on like a virus scanner interposed in the channel which is causing a multi minute delay while it accepts the message from one end but doesn't send it on to the other end.

So, first, you need to identify where the duplication occurred, in the surgemail logs find the message id in question and you will see if it was received multiple times or not.

Then go backwards to the source till you know where the duplication is occurring, then on both systems in question (the sender or mail client and receiving mail server) look for virus scanners and smart spam filters that might have caused an issue. And increase any timeouts you can see, in surgemail the timeouts you can increase are:

G_SEND_TIMEOUT
G_SMTP_CMD_TIMEOUT
G_SMTP_DATA_TIMEOUT

Now, the one most likely to help is the data timeout, but, it's a new one, you may need a special build to get that setting, let me know platform and we'll send a new build you can try which has that setting. With this type of fault I would set it up to about 20 minutes to see if it fixes it, e.g.
`g_smtp_data_timeout "1200"`

Lastly, some email clients will download messages multiple times by mistake, this is unlikely to be the problem but worth keeping in mind, particularly if the setting to 'leave message on server' is ticked in the mail client. Again the `msg*.rec` file should make it clear if this is a likely explanation as it will only show one message arriving.

AOL have blocked my server, what can I do?

AOL have some very strict policies regarding how much spam you can send them and your users can easily get your server blocked. The first thing to do is to go to <http://postmaster.aol.com> and open a feedback loop, this will let you know exactly how you are getting blacklisted, from there you should be able to kill the problem in SurgeMail. Often it is users that have setup redirection rules from their accounts to their AOL accounts.

You could prevent your users doing this with the following settings.

[g_forward_illegal](#) to="*@domain.com,*@domain2.com" apply="user"

will prevent users configuring forward rules to specified domains.

it will not prevent existing settings from working, if you want to find those try:

[tellmail_find_user](#) domain fwd *@domain.com
[tellmail_find_user](#) domain fwd *@domain2.com

SMTP session numbers growing, g_thread_max, g_smtp_max, g_pop_max limits

This problem has many causes, usually a broken virus scanner, or bad `dns_host` entry and orbs lookups causing a problem. If you just increase the limit without first understanding why the limit is being hit, you may well make the problem worse and hide the real cause of the problem while generating more obscure problems. So unless you

really expect that many sessions for some reason first read through this section and check the relevant status info.

If there are lots of pop/imap sessions then it may be an authentic problem or a disk IO problem.

In any case **first look at advanced 'status'** and examine the state of all the channels and how long they've been idle, if they are all in the same state then it's likely the name of the state will give you a clue as to what it's doing and what is mis configured.

If you are using a virus scanner other than avast, then we recommend you change to avast, the other free or third party scanners cause endless problems, they are fine for 'little' servers but when you are running a real server you should be seriously considering getting avast to avoid the headache's :-)

The other common problem is dns/spf lookups causing a problem to fix add this setting:

g_dns_lookup "true"

This is on by default in 3.7b and later builds.

If you are really getting hit by thousands of concurrent incoming email then first try reducing the timeouts:

```
G_SMTP_CMD_TIMEOUT "30"  
G_SMTP_DATA_TIMEOUT "62"
```

If that still doesn't help increase the limits g_thread_max and g_smtp_max, but not beyond 900 and 800 respectively and start lower (e.g. 500,400)

Read the next section 'only' if you've eliminated all issues above.

Increasing g_smtp_max and g_thread_max safely (too many files open)

Ok, first, read the above section, there is almost no chance you really need to increase these limits so chances are you need to fix the problems described in the above section, if you increase the limit without fixing the above problems then the problem will continue and it will be a lot harder to figure out the real cause.

To safely increase the limits do the following:

- In surgemail_start.sh set **ulimit -n 4096** (increased from 1024)
- Upgrade to surgemail 3.8c-5 or later
- Set limits, I suggest you still keep below 2000, **g_smtp_max "1800" g_thread_max "2000" g_pop_max "1500"**

NoSubmit errors on incoming email

This error may not mean a problem exists, it can occur normally when your system is probed. However it can also indicate a problem with the data stage. Basically it suggests 'something' is breaking the connection when the message body is being received.

Likely culprits are

Virus scanners (on your server, or on your gateway, or on the sending server) - Remove/uninstall these nasty things :-)

On windows it might be an MTU issue, Try "DrTCP" from dslreports (<http://www.dslreports.com/drtcp>) , decrease the MTU to 1024.

How do I move spam to a spam folder for all users

Ok, first, don't do this :-), in general this is a bad idea, if you have the surgemail spam settings correctly configured the users won't get spam so this won't be necessary, see

<http://netwinsite.com/surgemail/help/spam.htm>

But, there are some situations where this is worth doing, lets say you have an external spam filter tagging the messages, then possibly this might be a good idea. Do it as follows.

In the web admin tool click on accounts, go to the bottom of the page, and then click on the 'filtering' button

under 'Default user settings for this domain or global.

Add a rule to move messages into the folder you want if the header/tag exists, note you can add the 'if exists' check box so it will only do it if the user has created the specified folder.

Then you may wish to add an expire rule to expire the contents of this spam folder if it has messages more than 60 days old

(for each domain add)

expire_rule folder="Spam" age=60

How do I setup SurgeMail to be a backup server for a domain?

Let us use "mydomain.xx" as the example domain. You want to setup SurgeMail so that if the primary server that is hosting mydomain.xx goes down SurgeMail will accept mail for that domain and hold it until the primary server goes back online and then SurgeMail will deliver the mail to that server.

You need to make sure that you have a lower priority MX record for the domain pointing at your server to start with so that mail will be delivered to SurgeMail when the primary server is down. Then you just need to configure SurgeMail which is very easy.

You only need one setting.

[g_gateway](#) domain="mydomain.xx" to="ip of primary server"

The important thing is that you must NOT setup the domain on SurgeMail as otherwise SurgeMail will think that the domain is hosted locally and try and deliver the messages locally and they will of course fail as the users won't exist.

You can control the time period SurgeMail will hold these messages for the primary server before bouncing them with the

[g_retry_rule](#) setting, by default it will use the [g_retry_limit](#) setting which has a default of 48 hours, so SurgeMail will continue trying to send the messages to the the primary domain for 48 hours and then will bounce them.

In what order do the various parts of SurgeMail run? (g_virus_cmd, g_virus_filter etc etc)

Here is a short list of some of the functions in SurgeMail listed in the order they are run. g_virus_cmd is run first.

g_virus_cmd
g_filter_pipe
aspam
mfilter
g_scan_cmd

To convert a PFX file to a PEM file, follow these steps on a Windows machine:

1. Download and install the Win32 OpenSSL package from <http://gnuwin32.sourceforge.net/packages/openssl.htm>.
2. Create a folder c:\certs and copy the file yourcert.pfx into the c:\certs folder
3. Open a command prompt and change into the GnuWin32\bin directory:

```
cd %ProgramFiles%\GnuWin32\bin
```

4. Type the following command to convert the PFX file to an unencrypted PEM file (all on one line):

```
openssl pkcs12 -in c:\certs\yourcert.pfx -out c:\certs\cag.pem -nodes
```

5. When prompted for the import password, enter the password you used when exporting the certificate to a PFX file. You should receive a message that says MAC verified OK.

Copy the resulting file to the surgemail certificate file surge_priv.pem

SurgeMail crashes on 64bit linux every minute - libgcc_s.so.1 must be installed for pthread_cancel to work

Install the 32bit GCC RPM then 32 bit binaries will run on your 64bit installation.

Setting up ATRN service for a client

Define the mx record for 'example.com' to point to your mail server.

In surgmail.ini define:

```
g_atrn_port 366
g_atrn_server domain="example.com" user="fred" pass="secret"
g_relay_to "example.com" (so that surgmail will store mail for that domain)
g_retry_rule domain="example.com" hours="200" (keep messages for several days)
```

And open port 366 on your firewall if necessary and restart surgmail.

Then the client who's domain is 'example.com' should configure his server to use atrn username "fred" password "secret" to fetch pending email on your server.

How can I stop local accounts being compromised and used for spamming?

Limits to prevent guessing passwords and abusing a local account to send spam:

```
g_recent_bypass "127.0.0.1" # bypass limits per ip address
g_bad_login_ip_ignore "127.0.0.1" # bypass limits for bad logins
G_BAD_LOGIN_ALLOW "10" # Number of bad logins before blocking user
G_BAD_LOGIN_IP_ALLOW # number of bad logins before blocking that ip address

# limit users from sending out bulk email...
g_user_send_max max="2000"
g_user_send_warning "500"
G_USER_SEND_IP "true"
G_USER_SEND_WHITE "127.0.0.1,other known mailling list servers"
```

You can also check for weak passwords used by your users with the following command (run in the shell or command prompt)

```
tellmail test_weak
to find the worst accounts/passwords.
```

How do I restrict who a group of users can email?

In certain situations you may want to limit certain groups of users to not be able to send / receive mail directly from the internet.

eg in schools or in company environments

This can be done using g_access_group with an appropriate g_user_send_rule and g_user_receive_rule.
eg. accounts in the "local" group can only send to / receive from other mydomain.com accounts:

```
g_access_group group="local" access_pop="*" access_imap="*" access_smtp="*" access_incoming="*"
g_user_receive_rule group="local" from="*@mydomain.com"
g_user_send_rule group="local" to="*@mydomain.com"
```

Bounce blacklisting due to responders friends etc ironport filtering or spamcop

Some RBL systems (like spamcop) blacklist servers for sending bounces. One could argue if this was valid or not, but they do it, and so we must cope with it

In general surgmail doesn't send many bounces compared to most servers but if you find your are being blacklisted due to this issue then consider using these settings:

- 1) Turn on the default recommended spam settings using the 'config' checker in the web admin tool
- 2) Set

```
g_spam_block "true"
```

- 3) If you have a front end mail server, or low priority mx hosts, then remove them, so mail goes directly to surgmail if possible.

4) If problems still persist then set

```
g_friends_check_spf "true"  
g_responder_safer "true"
```

5) If problems still persist, panic :-)

554 Failure tcp_read_dot

These errors indicate a fault external to surgemail, usually virus/spam scanner on the sending users pc is the problem. But let us know if you can't find the fault. Other potential causes include faulty network cards or random network failures. Or a corrupt email message and faulty sending email server.

You can identify more about the problem by testing as follows:

- Get the problem sender to send with a different email client, and to send different sized messages with different contents.
- Get the problem sender to send the identical message/attachment via another email service (e.g. gmail)
- You can turn on extra logging at your end using `g_log_tcp_read "their.ip.address"`, this will give a detailed log of what your server received before the connection closed. (see `tcp.log`)

554 Failure tcp_read_dot

This message means surgmeail never got a 'data' command from the sending server. So the sending server never actually tried to send an email message, this is most common if all the recipients are rejected, or if a virus or spam filter at the sending users pc rejects the message before it can be sent. It also happens when the sending mail server/client is just trying to find out if the recipient is valid.

It has been seen once with "Symantec AntiVirus 10.0.0.359 running the Internet E-mail Auto Protect. Disable the feature."

Customer Support

Please contact NetWin if you need any assistance.

- [Contact SurgeMail support](#) (Please use this address first)
- [Problem Escalation Form](#)
- Fax us on +6463537359 (Please try using Problem Escalation Form before faxing)
- [SurgeMail Customer Forum](#)

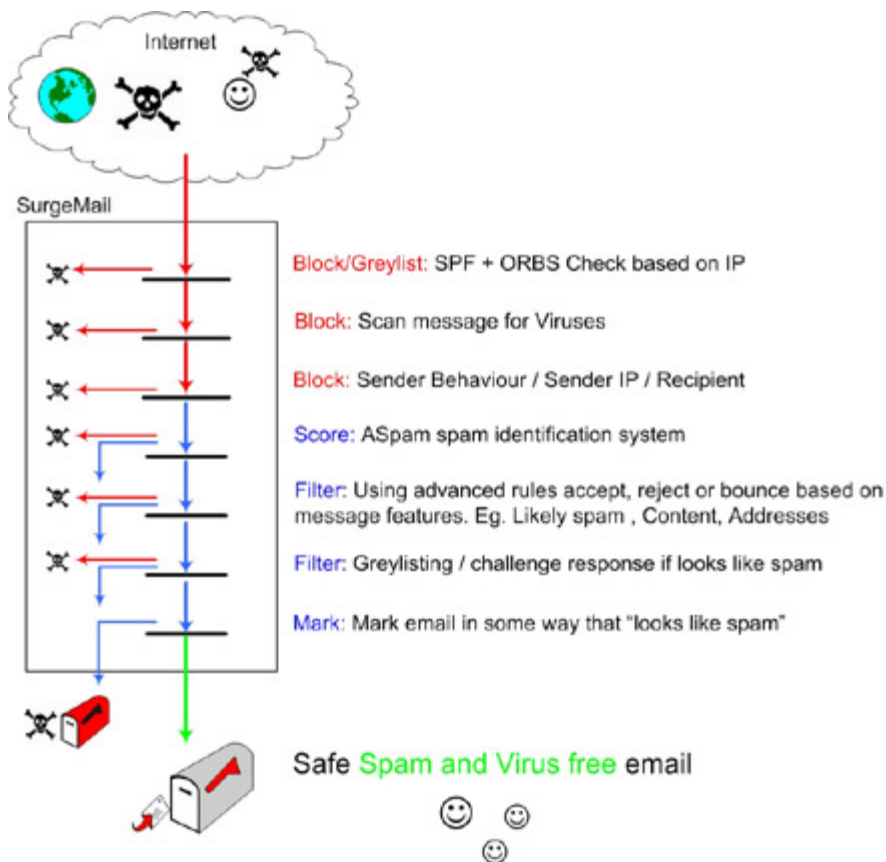
Please contact sales@netwinsite.com to purchase [extended support](#) options.

SurgeMail Spam and Virus protection

The best spam prevention techniques on the market today use SPF in "strict mode" (and to a lesser extent mechanisms such as ORBS) to only accept mail from someone confirmed to be who they say they are, combined with trainable message content based filtering.

For essential information on spam prevention using surgemail see [notes on stopping spam](#) and [spam prevention guide](#).

SurgeMail offers advanced features to identify undesirable spam email, block virus infected mail and prevent abuse of your mail server by spammers. Incoming mail is scrutinised as shown in the diagram.



SPF + Open Relay database check

SPF ([Sender Policy Framework](#)) allows you to check that someone sending you mail is who they say they are. SurgeMail has unique features that combines SPF with grey-listing and the allow mechanism to ensure there is minimal impact on valid mail. In particular, using greylisting SurgeMail will automatically start accepting mail from real mailservers, and using the allow mechanism any valid end users whose message has been blocked is able to add their IP address so that subsequent messages are accepted.

It is important that you have SPF settings correctly configured. If they are not correctly configured you may conclude that SPF is "ineffective" or "stops too much real mail". Make sure that you follow the [spam prevention guide](#).

SurgeMail's integrated and flexible [open relay database checking](#) can be used to enforce a server's blacklisting or whitelisting in one or more ORBS databases. In addition, this can be used to mark messages with a header which can then be taken into account in the ASpam "SpamDetect rating" calculation. An ORBS database is simply a DNS server that returns a positive response if a server is listed in the database. A variety of services are available online that can maintain blacklist databases. Normally you would maintain your own whitelist database that overrides the blacklist listings.

Scan messages for viruses

SurgeMail has a variety of mechanisms for integrating with [external virus scanners](#).

- [Integrated and efficient Avast scanner](#) (windows / linux only)
- [Efficient external scanners](#)

- [External SMTP scanner](#) ("virus wall")
- [Arbitrary command line scanner](#) (deleting message or return code)

Any of these mechanisms can be used but it is recommended that Avast is used as this is closely integrated with surgemail, is efficient and is less prone to errors under load.

ASpam spam scoring system

SurgeMail has built in support for [ASpam](#). This is a message "spamminess" scoring system based on the sum of the following:

- Locally customisable rule database maintained by netwin staff	Approx 60% accurate on common spam
- Auto training database of recent messages that " look like spam " based on poly and multi symbol statistical word matching.	Approx 90% effective if no local training is done, approx 99% effective if local training is done.
- Auto training database of recent messages that " look like spam " based on message parameters such as URL and content	Approx 40% effective if no local training is done, approx 99.5 effective on trained data.
- Catcher addresses that should never receive genuine mail	If mail is received on these addresses it a known a spammer.
- Optional modification of scoring based on ORBS and SPF checks	

The auto training databases consist of a base set of rules maintained at netwinsite.com combined with local training based on messages submitted by the users of your system as uncaught spam or as a false positive.

Based on this "SpamDetect score" messages can be filtered at a serverwide level or at a per user level allowing individual users to fully customise their filtering setting up a totally customised "personal antispam policy" based on their chosen level of spam 'tolerance'. One of the more useful techniques is to use the Friends "Challenge / Response" only on messages that look almost certain to be spam.

Custom Filtering rules and Advanced mail rules

A variety of mechanisms exist in surgemail to setup custom filtering rules, exceptions to spam filtering and custom mail handling policies at a per user or serverwide level.

Examples of what the **per user** filtering allows users to is:

- take action (accept / reject / forward / move to special folder) based on message parameters
- take action (mark/bounce/move to folder/etc) on messages that look like spam
- send challenge response requests to certain messages (look like spam/ from certain domains/etc)

At a **domain** or **serverwide** level customised policies can be setup for:

- configuration and customisation of each spam prevention mechanism and default
- mail forwarding, redirection, archiving options
- default per user rules that are applied if user has no rules defined
- [mfilter](#) based advanced mail filtering

Other techniques

Some of the other surgemail techniques that can be used in preventing spam and virus infected emails getting to end users include:

Friends only system

The friends only system is a **challenge response** system allowing users to opt to receive messages only from friends. Non friends are automatically questioned to determine if they are human. All mail from non friends is held pending on the server until the user decided what to do with it. Status reports are sent to the user on a regular basis to provide information on the Friends system and any mail pending delivery.

Further information on [configuring the friends system](#).

Message attachment renaming

SurgeMail can [disallow](#) or [rename](#) certain message attachments as a basic antivirus tool.

Sender behaviour limitation

SurgeMail has many configuration options to directly block or tarpit users or servers identified as abusing the your mailserver. Some settings are:

- [g_deny](#) - Deny users from some IP ranges
- [g_ban_rcpt](#) - Ban any matching RCPT TO: envelope
- [g_ban_from](#) - Ban any matching MAIL FROM: envelope
- [g_ban_helo](#) - Ban any machine that gives a matching 'helo' string
- [g_tarpit_max](#) - Max recipients per hour from one IP
- [g_bomb_max](#) - Max messages to a single address per hour
- [g_max_bad_to](#) - Max bad recipients in a row
- [g_con_perip](#) - Max simultaneous connections per IP

Spam Prevention using SurgeMail Features

- [How to enable SPAM handling](#)
- [How users can report/train spam](#)
- [How it works](#) (in brief)
- [Technical details on how SurgeMail stops spam](#)
- [List of recommended settings with notes](#)
- [Settings to stop spam of the form from=to \(where the sender pretends to be the recipient\)](#)
- [Stop hackers from sending spam from your server!](#)
- [Frequently Asked Questions](#)

NOTE: Turning on the recommended settings WILL NOT block email from servers without SPF. This seems to be the number one confusion. Many customers are reluctant to turn on the settings recommended as they fear their server will then bounce all email from badly configured mail systems, this is NOT the case, please try the recommended settings as a starting point!!

How it works

Spam prevention has gone through many changes over the last few years, initially people tried to filter spam based on the 'content' although this worked well initially it soon started to fail as spammers adjusted their spam. The focus of successful spam prevention is based on a multi pronged attack, where the 'source' of the message is verified in various ways, and the content of the message is checked, and then finally the 'friends' system catches and automatically deals with any messages that still get through while it also automatically white lists known associates.

For more information see the [detailed technical description](#) of how SurgeMail stops spam when correctly configured.

How to enable SPAM handling

1. Upgrade to the latest stable release
2. Press the 'config checker' button in the web admin interface and turn on the settings it suggests
3. Either set G_FRIENDS_DEFAULT_MODE "smite" or "silent" or "list"
 - smite = If message is placed in 'spam' folder tell 'sender' and give them a url to allow delivery of their message
 - silent = If message is placed in 'spam' folder do not tell the sender
 - list = Maintain friends lists for whitisting but deliver all mail (including spam) to the users inbox
4. Tell all users to individually turn on/set/adjust their friends options in the user self admin settings.
5. Advise your users how to turn on/off user configurable options (this is done in the user self admin pages or via options in surgeweb)

How users can report/train spam

1. IMap users can drop spam messages into the 'Spam' folder
2. SurgeWeb users can click on 'Is Spam'
3. POP users must forward spam to 'isspam@yourdomain.com' (this is the least accurate method as the server must then untangle the sent message, options 1,2 above are better)

List of recommended settings with notes

g_orbs_list name="zen.spamhaus.org" action="stamp" stamp="zen.spamhaus.org , ip=||remoteip|| "

This setting tells surgemail to check the IP address with an RBL service (in this case spamhaus) This setting improves the spam scoring features. Please check

<http://www.netwinsite.com/surgemail/help/rbl.htm> for more information on RBL's.

g_verify_mx_skip "10.0.0.2"

This setting should list your other MX hosts (low priority mx host). However our general recommendation is to REMOVE low priority mx hosts entirely as they serve no useful purpose and will tend to allow spam through your system.

`g_spam_allow "10.2.192.98-117"`

This setting lets you list the ip addresses of known trusted hosts.

`g_spam_subject "4"`

This setting adds **** to the subject of messages that score more than '4'.

`g_spam_userconfig "TRUE"`

This setting lets the users change their own spam settings.

`g_spam_internal "true"` ("Enable ASpam" setting in user interface)

This turns on the asspam scoring system.

`g_spam_catcher "fred@your.domain"`

This setting is used to train the asspam filter with spam that comes to special email addresses on your system, place these email addresses on your web pages so that spammers will accidentally train your system for you :-)

`g_url_enable "true"`

This adds some url scoring using a netwin provided database that is updated every few hours, you should also use SURBL as well.

`g_vanish_bad_bounces "TRUE"`

This gets rid of bounces that didn't originate from your server.

`g_verify_smtp "TRUE"` (Probably not needed when using spf)

This setting checks if the connecting smtp server is open on port 25. The spam scoring is adjusted if the test fails.

`g_spf_mode "strict"` **(absolutely essential!!!)**

`g_spam_block "true"` **(absolutely essential!!!)**

`g_spam_allow_known "true"` (this allows more spam through but cuts down on rejections)

`g_spam_grey_dflt "false"`

`g_spam_grey_dflt_bad "true"`

`g_spam_grey_bounce "10"` (explained below)

These settings turn on SPF see <http://netwinsite.com/spf.htm>. In addition the `g_spam_block` setting makes it actually block all the spam that fails spf tests. However to reduce impact the grey settings mean that failures are grey listed, and only fully blocked if grey listing fails, or if too many messages arrive within a short time period (1 message)

`g_surbl name="multi.surbl.org" stamp="sc.surbl.org,ws.surbl.org,phishing,ob.surbl.org,ab.surbl.org,jp"`

This setting is critical to spam detection, the surbl database is used to detect urls that spammers are trying to promote.

`g_spam_grey_bounce "10"`

This setting lets surgemail bounce a message that looks 'spammy' if it failed some spf tests but got past the grey listing mechanism. This cuts down on spam but does often bounce real emails (it uses an allow bounce so the sender can fix it)

It's probably better for individual users to use their friends settings instead.

This used to default to 3 and still is on any version prior to "SurgeMail Version 3.8". The new default value is 10 which lets more spam through, but reduces accidental bounces. We now recommend a value of '10' unless you are happy with some real legit email bouncing.

Settings you SHOULD REMOVE.

We often find problems occur when non standard settings or obsolete settings have been turned on, here are the main culprits you should remove. These will break the normal spam prevention occurring and or cause massive complaints from users due to bounces.

```
(remove) g_spf_default_noblock "TRUE"
(remove) g_spam_grey "TRUE"
(remove) g_spam_grey_dflt "TRUE" (optional)
(remove) g_spam_allow_disable "TRUE"

(remove) g_badfrom_check "TRUE"
(remove) g_badfrom_stamp "TRUE"
```

Frequently Asked Questions FAQ

Can I avoid backscatter from friends?

Yes use this setting

G_FRIENDS_CHECK_SPF "true"

What are the recommended best practise techniques to avoid spam on my server?

See the list of settings above, primarily you want SURBL, RBL's and SPF (in strict mode with the g_spam_block turned on). Also avoid using front end filter systems as these will prevent the best spam features in surgemail working. And suggest users turn on 'friends' with a friends bounce level of about 4.

Doesn't SPF rely on the senders creating spf records ?

No, in strict mode surgemail makes up an spf record for all incoming domains so it works for everyone. When the made up spf record fails (which is rare) surgemail then provides other checks and mechanisms so real email can still get through.

Is there something else I should be doing to prevent spam, why do I get so much when other people get none?

Although these mechanisms can stop almost all spam, there is another way to get rid of spam, and if you use it, then you can adjust the filters to be very 'forgiving' so that real messages are never caught by them. So here's the trick, the BEST way to avoid spam, is to change your email address! and keep your new email address private:

- Never put an email address on a web page, use a form instead (our free [easyform](#) product for example)
- Never post to a public news group except through a system that hides email addresses (See [SurgeNews](#))
- Never join a mailing list. Instead use an RSS feed (See [SurgeNews](#)), or a special second email account.

What are the likely side effects and implications of using these measures?

You will bounce some real mail messages and because some people don't read the bounce messages they will actually fail to respond correctly to get past the automated spam prevention. The above settings only require responses from about 1-2% of people so most mail gets through without any trouble, but a small percentage will be bounced and if the user sending doesn't respond then the message will fail to be delivered. This results in a loss of about 0.1% of messages, much lower than letting humans do the filtering, but still not perfect.

How do I measure how effective these techniques are? (my manager needs a report to justify costs)

In the advanced status section in surgemail there is a 'spam' section, this has figures on the various filter hit rates, it's a little hard to interpret but it gives a fairly good idea of how much spam has been blocked.

How are false positives handled? Each email is important to me, and I must avoid false positives at all costs, how can I monitor email identified as spam until I am confident that the system has no / minimal false positives?

With SPF and friends false positives result in some form of bounce, the user sending the message must then respond to the bounce to get their original message delivered. (With SPF failures they must resend, with Friends they need not). You will only loose messages when the person sending to you does not read the bounces. From the user web interface you can search through all the bounces manually and release messages pending confirmation via friends, and fix SPF failures.

How can spam that was not caught be submitted (by users)? and how do users/admin get feedback that their submissions are actually doing something?

You or any user can send messages to `isspam@your.domain` or `notspam@your.domain`, this will improve the scoring in future. From the managers web admin pages for spam you can also paste in a message and get it analyzed, or trained. This process should not be over emphasized, it is good for fine tuning the filters slightly but it is not at all critical that you submit every failed message or every false positive. The messages can be sent as attachments or redirects, it doesn't matter much which is done as the system is forgiving. If a messages is sent to the wrong training address, just resend it to the other address to nullify the training.

How should I as a user configure my spam controls on my email. There seem to be several ways of configuring filters + friends + spam/spf etc to work together. Why should / whould I not use a particular combination. Are there particular things that I probably should not configure?

This is very important, if you get 'lots' of spam and want to get none.

- Set the SPF setting to BLOCK
- Set friends mode to anything above '2' stars

If you get a small amount of spam but want to get rid of 'most' of it, without much risk of ever bouncing a real message:

- Set friends mode to anything above '6' stars

Are there any significant performance effects? (on 100 / 5000 / 100000 user system) Both in increased load that these measures put on system resources (disk / cpu / open channels / resposiveness etc) and reduced load by not having to deal with spam. How can I measure these effects?

Not really, the spam system in SurgeMail is very efficient and the SPF features and vanish bad bounce settings do reduce real load on heavily spammed servers, so the spam prevention tends to result in a slight performance improvement, and reduced network bandwidth usage.

We do STRONGLY recommend the use of the AVAST virus scanner product as it is enormously more efficient than some of the free unix command line scanning utilities that you can use with SurgeMail (mainly because it does not get activated for each scan as it's part of the server)

Also using external spam checking systems (which you can do if you really want to) is also strongly discouraged, these generally won't increase your filtering accuracy but will badly affect performance.

I want to counter some rules in ASPAM - for example NakedCR.

ASPAM's filter rules are stored in `aspam_mfilter.txt`, you cannot edit this file as it is updated regularly so any changes you make will be overwritten. You need to edit the file `local.rul` where you can add your own rules.

```
if (isin("X-NakedCr","body")) then
  call spamdetect(0.1,"NakedCR")
end if
```

In general, look through `aspam_mfilter.txt` find the rule and then write the same rule in `local.rul` but with a negative score to cancel the scoring in `aspam_mfiler.rul`. The string/reason in `local.rul` must be exactly the same as the string in `aspam_mfilter.rul` for the rule to override the first one.

New settings that we hope to make recommended in future.

These settings are typically new settings only available in the **latest beta builds**, and **may be unstable**, but once stable we expect to become recommended settings so you might want to experiment with these.

g_domainkeys_check "true"

Checks incoming email for signatures and if found verify, this will help avoid bounces from domains that use domainkeys instead of spf.

g_domainkeys_sign "true" (see note below)

Use the web admin to create your finger print and then save in your dns first.

g_spam_share "true"

Use and contribute to shared whitelist database via netwinsite.com to avoid spf bounces for well known sites that are not spammers but fail spf tests.

Integrating external SPAM filters

You can in addition to the normal surgemail spam features run an external spam filter which is a command line program that examines the message then returns non zero numbers if it thinks the message is spam. This can then contribute to the SurgeMail score for that message.

We recommend this external filter, it's a reasonable price and seems to work reasonably well:

<http://www.armresearch.com/message-sniffer/> we are keen to hear feedback from anyone using filters like this.

These settings require SurgeMail 3.8-20 or later, email support@netwinsite.com if you need this build to try this new feature.

Surgemail.ini setting:

Replace the weird code with your temporary license that they provide and correct the path to the binary.

```
G_SPAM_CMD "c:\surgemail\snfrv2r3.exe xnk05x5vmipeaof7 $FILE$"
```

Add some rules to your local.rul file to process the resulting header:

```
if (isin("X-SpamCmd","Is Spam")) then
    call spamdetect(5,"ArmResearch")
end if
if (isin("X-SpamCmd","Not Spam")) then
    call spamdetect(-3,"ArmResearch")
end if
```

Forgeries of the form From=To

There has recently been an increase in spam where the From and To headers are set to be the same as the user. To block this type of spam ensure you have done the following steps

- Install the latest version of SurgeMail (3.9h-61 or later)
- Turn on the Recommended settings using the web admin tool config wizard.
- Set **g_from_stamp "true"** and **g_from_noforgeme "true"**
- We also recommend you have an SPF entry for your domain in your dns server, and set **g_spf_enforce_local "true"**
- In version 4.0b-19 and later you can use the command "tellmail scan_friends" to detect users who have accidentally added themselves to their own list of friends (this is not possible to do in new versions but many users did it before we made it impossible). In addition it's possible to remove those entries with the 'repair' switch for that command.

Optional settings to stop more spam...

Some of these are a bit 'strict' so use with caution depending on your tastes...

You may wish to try this setting, it will black list any ip address that is the source of a issspam training event for an hour or so, this is most useful with your catcher addresses as it means any spammer who sends to your spam catcher will find themselves blocked from sending any email to your server for an hour or so. You may need a whitelist for a few large sites to avoid issues with deleted users causing a large mail server to get blocked. Hence the g_black_white setting given as an example...

```
g_black_isspam "true"
```

```
g_black_white "1.2.3.*,*ebay.com"
```

You can tell surgmail to try and guess the language of each message, you can then set for any account in your spam settings the language you expect, if you get messages that are not in your language (e.g. english) then the message will be assumed to be spam until proven otherwise (So it goes to your friends pending folder), this will reduce spam significantly for those of us who really only speak one language :-). Be warned it does not always guess correctly, and is more likely to be wrong with non english messages.

```
g_spam_lang "true"
```

Stop hackers from using your server to send spam

Hackers are now probing mail servers all the time to find email accounts with 'easy' passwords, they are probably already probing your server. They will break in if you have any accounts with simple passwords. So on a large server its not a question of 'if' your server will be hacked, it's really more a question of when. You need to make it harder for the hackers, and you need to be ready to detect the locally hacked account and shut it down quickly before your reputation suffers!

20-30% of users who have their accounts hacked 'won't change their own passwords even once they are told by someone that they have been hacked. So don't expect your users to take action themselves :-)

Here are some things you might consider to help stop this occurring, and to help identify it when it does occur.

```
# List top senders (to identify the account that might have been compromised)
tellmail send_top
```

```
# Find any local accounts with really really obvious passwords!
tellmail test_weak
```

```
# Login guesses per IP before it is automatically and permanently locked out. Use tellmail unlock ip.address to fix...
G_HACKER_MAX "10"
```

```
# If hacker attempts to login to one of these then the ip is instantly locked out. (Don't use accounts that exist)
G_HACKER_POISON "root@*,administrator@*"
```

```
# Only allow smtp logins if the user has previously logged in via imap/pop from the same address
G_SAFE_SMTP "true"
```

```
# Max messages an authenticated user can send per 30 minutes, e.g. 5000
G_SPAM_USER_MAX "2000"
```

```
# Max outgoing messages per ipaddress/return path pair, 30 minutes, e.g. 5000
G_SPAM_FROM_MAX "2000"
```

```
# Detect local users sending 'spam like' email and send a report to the manager.
G_OUTGOING_N "5"
```

```
# White list for people you know send mail that looks a bit dodgy.  
G_OUTGOING_WHITE "bob@here.com,1.2.3.4"
```

```
# send manager an email if a local user sends more than 300 message in a day...  
G_USER_SEND_WARNING "300"  
g_user_send_ip "300"
```

SECURITY_SUFFIX - Make logins fail unless the user knows the suffix

One other method to protect your server is to make the login username different from the email address. You can do this on a per domain level, lets say you have a domain MYDOMAIN.COM and you want the users to login with username=JOHN@SECRET.MYDOMAIN.COM

```
g_from_rewrite was="*@secret.mydomain.com" to="%1@mydomain.com"  
g_from_rewrite_header "true"  
g_from_rewrite_sender "true"
```

```
vdomain name="mydomain.com"  
security_suffix "secret.mydomain.com"  
...
```

note: I don't recommend using the security suffix on most systems, it is the most complex of the settings and the most likely to cause disruption, it does have benefits but I would only use it for small sites where extreme security measures are justified.

Realtime Blackhole Lists (RBL's)

One of the best ways to fight spam is to use RBL's. RBL's are lists of servers that usually you don't want to talk to, sometimes they are lists of servers that are open relays or they are lists of servers that are proxies or maybe lists of servers that are dynamic ip's. There are many different types of RBL's out there so you might want to do some investigation before you deicde which ones to use. The idea is that when a server connects to your server SurgeMail will then check the RBL to see if the connecting server is listed, if it is then we can simply drop their connection or we can stamp the message to say its listed on a RBL and increase the ASPAM score. Using RBL's can dramatically decrease the amount of SPAM coming into your system and we highly recommend using them.

Here are some RBL's you can use. Please note that you should double check with their website to make sure these are still operating.

Name of RBL	What to enter in SurgeMail (name section)	Response Code	General Information on RBL.
spamhaus	sbl.spamhaus.org	127.0.0.2	Very well known RBL, well recommended. Direct UBE sources, verified spam services and ROKSO spammers More info here on SBL list. http://www.spamhaus.org/sbl/index.lasso
spamhaus	xbl.spamhaus.org	127.0.0.4-6	Illegal 3rd party exploits, including proxies, worms and trojan exploits More info here on XBL list. http://www.spamhaus.org/xbl/index.lasso
spamhaus	zen.spamhaus.org	127.0.0.2 127.0.0.4 5 6 127.0.0.8	If you want to use both SBL and XBL and the new PBL then you can just enter this into SurgeMail. More info here on ZEN list. http://www.spamhaus.org/zen/
Domain Name System Real-time Black List (DNSRBL)	dun.dnsrbl.net	127.0.0.2-9	List of IP addresses of machines that are either direct SPAM sources or Dial-up (dynamic address) pools which would never be a source of non-spam messages.
RFC Ignorant (Whois)	whois.rfc-ignorant.org	127.0.0.7 or 127.0.0.5	List of IP's that do not comply with RFC's. (Careful about using this one) ?
Not Just Another Bogus List (NJABL)	dnsbl.njabl.org	127.0.0.2-9	List of known and potential spam sources (open relays, open proxies, open form to mail HTTP gateways, dynamic IP pools, and direct spammers).
Spamcop	bl.spamcop.net		

There are plenty more out there, but the above ones are well known and will probably do the trick
To add them into SurgeMail, click Spam control, then scrol down in the right frame until you find RBL settings.
You will need to click on the advanced mode link to view all RBL settings.

[Bulletins](#)

[Migration](#)

[Spam control](#)

[Antivirus](#)

Status & Reporting

[Status](#)

[Mail queue](#)

[Queue stats](#)

[Log](#)

RBL settings : RBL lists can be used to outright block listed systems, or modify the SpamDetect score as used by ASpam. The recommended mode is stamp.

[Realtime Blackhole Lists \(RBL's\). action=deny,accept,stamp](#)[Edit RBL's](#)

[Realtime Blackhole List, exception list of IP addresses](#)[Edit Rules](#)

Save Changes

(9 additional settings in [advanced](#) mode)

Once you have clicked on Edit RBL's

Find config setting:

☐ Manager page

☐ Common

☒ Global settings

☐ General

☐ Relaying

☐ Redirecting

☐ Spam control

☐ Filters

☐ Security

☐ Multiserver

☐ User

☐ Webmail

Realtime Blackhole Lists (RBL's), action=deny,accept,stamp

Name	Action	Stamp
sbl-xbl.spamhaus.org	deny	Your ip remoteip is listed in the
bl.spamcop.net	stamp	Listed in SPAMCOP

Save Changes

Help

So under the name section you add the domain of the rbl (eg bl.spamcop.net), then what action you would like to take (deny, accept, or stamp) and then in the stamp section you can add the stamp.

deny = connection is banned and sending server is sent the stamp message.
stamp = message will be allowed through but it will be stamped with stamp you set. The stamp is a message header and should normally start with X- eg X-RBL: Listed in SPAMCOP (||remoteip||)
The RBL's are processed in the order they are listed and if the sending server is found on one of the RBL's the rest will not be checked to save processing power.

There are servers that you might not want to ever risk being denied, sometimes servers can accidentally get themselves on RBL's or the RBL's can add servers by mistake at times. In the first screen shot you can see the third option(exception list of IP's) allows you to add IP's that will never be checked by SurgeMail.

If you click on the advanced mode in the web admin you can look for (do late disconnect - [g_orbs_late](#)) This means that the your users are allowed to authenticate first and then the RBL checks are done, this means that if your users are on a RBL they will still be able to send messages through your server. This can also be used with the setting [g_spf_skip_to](#) which allows you to add recipients that will be bypassed for RBL checks, so you might add postmaster in here as everyone should be able to send to postmaster.

Editing surgemail.ini directly

For those that prefer to edit surgemail.ini directly here are the settings and some examples

```
g\_orbs\_list name="zen.spamhaus.org" action="deny" stamp="Your ip ||remoteip|| is listed in the spamhaus RBL http://www.spamhaus.org"
g\_orbs\_list name="bl.spamcop.net" action="stamp" stamp="Listed in SPAMCOP"
```

RBL Exceptions:

```
g\_orbs\_exception "ip,ip,ip"
g\_orbs\_late "true" - This makes the RBL checks happen after the authentication phase and allows you to also use g\_spf\_skip\_to.
g\_spf\_skip\_to "fred@mydomain.com" - This will make SurgeMail not use the RBL if the message is going to fred@mydomain.com and you also have g\_orbs\_late "true" activated.
g\_spf\_skip\_from "fred@anotherdomain.com" - Will make SurgeMail skip RBL checks if the from address matches this, must have g\_orbs\_late "true" activated.
g\_spam\_allow\_rbl "true/false" - Give unblock message to RBL bounces too. Make sure you read documentation on this setting before setting to true!.
```

Misc. RBL settings

```
g\_orbs\_force "true/false" - Force RBL check even if g\_allow\_ip matches this ip number.
g\_orbs\_timeout "seconds" - Seconds to wait for RBL lookups, default is 10 seconds.
g\_orbs\_report "true/false" - Use this setting to test your own ip addresses, as soon as one is found in orbs you will be sent an email to alert you.
g\_orbs\_check\_all "true/false" - Keep doing lookups even if found in a RBL.
```

HostKarma junkemailfilter.com

This is like an RBL but some responses are good, and some are bad. So you use a rule like this:

```
g\_orbs\_list name="hostkarma.junkemailfilter.com" action="stamp"
stamp="127.0.0.1=hostkarma_white: accept~127.0.0.2=http://ipadmin.junkemailfilter.com/remove.php: deny~127.0.0.3=hostkarma_yellow~127.0.0.4=hostkarma_brown~127.0.0.5=hostkarma_nobl~127.0.1.1=hostkarma_quitok~127.0.1.2=hostkarma_noquit"
```

Note:
[g_relay_allow_ip](#) "ip" allows users to bypass RBL checks, this behaviour can be stopped by using the setting [g_orbs_force](#) "true"

Adding scoring to ASPAM when found in a RBL

Instead of just outright denying, you can set to stamp mode and then use those stamps to add scoring to ASPAM. The argument for using this method is it gives the end user more control and also adds a bit more reliability as you can set SurgeMail so it will only reject messages if found in a certain number of RBL's instead of just one.

So if we have:

```
g_orbs_list name="bl.spamcop.net" action="stamp" stamp="Listed in SPAMCOP"  
g_orbs_list name="zen.spamhaus.org" action="stamp" stamp="Listed in zen.spamhaus.org"
```

Then we would edit local.rul in the surgemail directory and add this to it.

```
if(isin("X-ORBS-Stamp", "Listed in SPAMCOP")) then  
    call spamdetect(6, "Senders ip was found in SPAMcop RBL")  
end if  
if(isin("X-ORBS-Stamp", "Listed in zen.spamhaus.org")) then  
    call spamdetect(6, "Senders ip was found in zen.spamhaus.org RBL")  
end if
```

The header that is always added is "X-ORBS-Stamp" so you always check against that.

The above will add 6 points if the senders ip is found in spamcop RBL. By default when SurgeMail finds a sender's ip in a RBL it doesn't bother checking the rest of the RBL's you have listed. In this situation it can be useful to make SurgeMail keep checking the other RBL's so that if the sender is found on more than one RBL it will increase the scoring and lessen the chances of a false positive and increase the chances that the message will be detected as SPAM due to high scoring. You can make SurgeMail do this with the following setting.

```
g_orbs_check_all "true"
```

So with this setting, if the above sender is found in both spamcop and spamhaus the message will have a total score of 12 added to it.

Server Status

SurgeMail server status information is available in two ways:

- from the command line using "tellmail status"
- using the Web Admin Interface - Status

The status display shows you information on the current status of SurgeMail and some summary statistics gathered since the server was last restarted. The most useful information is displayed in the first section:

Uptime

- Uptime - The hours/minutes since the server was started.
- Total bytes in / out - Bytes transferred since started

Message Delivery Statistics

- Connections - Total SMTP connections made (this includes tellmail command connections)
- Bad Connects - SMTP connections that were dropped as bad before a thread was created for the incoming connection (in particular g_thread_max, g_deny settings)
- Accepted - Messages accepted for SMTP delivery
- Stored local - messages delivered locally
- Sent on - messages delivered remotely
- Bounced / Dropped - messages rejected
- Smite - Status of optional SmiteSpam system
- Total in / out - Total messages received and sent
- Table showing delivery timing

Note: If you are wanting to track particular messages delivery this should be done in [Searching the log files](#).

Mail Queue Statistics

- Que startup - messages in the queue on startup
- Memory Que -
- Still in que -
- Unsent yet -

Mail Mirroring Statistics

- Mirror out - Status of mail sent to mirror server
- Mirror in - Status of mail received from mirror server

Internal SurgeMail Statistics

The following is information typically of use to NetWin support staff to see what is going on inside your server.

- Channels - Status on all protocol communication channels
- Threads - Status of all internal thread
- Stored local - messages delivered locally
- Mutex timing
- Function timing

Queue Status

Shows all messages in the in memory queue.

Queue Analysis

Analysis of messages in the mail queue.

Searching the log Files

SurgeMail generates two log files of general interest:

- msg.log (surge\mail\recYYMM\msgYYMMDD.rec) = permanent record of mail received, bounced, delivered, etc
- mail.log (surge\mail\mail*.log) = temporary record of almost everything that happens inside SurgeMail . These are rotated on restart or when they get too big (typically several hours, but if server load is high this may be as small as several minutes)

Log file searching

With this form you can do some quite advanced log file searching and troubleshooting. The search page has several ways it can be used:

1. The default behaviour is that the complete last part of each log is displayed to the size of the "limit results to" field.
2. You can manually search for specific things
3. You can drill down from the msg.log entries to mail.log entries
4. You can use the sample searches as supplied at the bottom of the page
5. You can click the suggested highlighting possibilities at top of displayed log items
6. Complete custom searches using the advanced page

These are explained in more detail below for now let us look at some examples.

The msg.log entries are formatted as: day, time, [queuenumber], action, and a variety of more detailed information

eg.
 19 14:26:27 [2] Received 127.0.0.1 <test1@orion> <test3@lap> 588 <1@test> "In Queue"
 19 14:26:27 [2] Stored 127.0.0.1 <test1@orion> <test3@lap> 588 <1@test> "Stored locally"

The mail.log entries are formatted as: day, time, log entry type, thread number, variety of more detailed information

eg.
 08 12:25:29.06:Info:2120: SurgeSMTP 1.5d, User connected (210.86.15.138) (10.0.0.5)
 08 12:25:29.07:Info:2120: smtp:[210.86.15.138] In: HELO mta5-rme.xtra.co.nz

By clicking on the thread id or date fields you can drill down to get more specific information related to a particular message or thread. By clicking on the day/time fields you are zooming out to get a display of all things happening at this time.

Using the search page

Searches consist of text to search "for", particular text to be "highlighted" and text to be "excluded". These are all comma separated lists of wildcard search terms that get combined.

eg to search for anything related to "fred@mydomain.com" OR "joe@mydomain.com" but NOT "alfred@mydomain.com" with joe's stuff highlighted you could use:

```
Search for: "userfred@mydomain.com,userjoe@mydomain.com"
highlight: "userfred*"
exclude: "alfred@mydomain.com"
```

So for multiple entries using comma separated search terms::

```
- "Search for:A,B" means: A or B
- "Limit to:A,B" means: A and B
- "Search for:A,B" + "Limit to:C,D" means: (A or B) and (C and D)
- "Search for:A,B" + "Limit to:C,D" + "Exclude:E,F" means: (A or B) and (C and D) and NOT (E or F)
```

In addition the search can be limited to a certain amount of output and log text in certain files. (See right hand side of the search page) You would typically just limit to a certain amount of output from the last one log file. But it is also useful to select a small amount of output from multiple log files and to actually limit the output by searched for text.

eg: to limit search for anything related to thread with id "12345" AND "over a timerange between 12:45 and 12:49 on the 15th of the month" in any of the mail.log files the following limits could be used:

```
Limit results to : 1000K of each All files
and limit to: ":12345:,15 12:4[5-9]"
```

Wildcards that may be used are *, ? and [].

(eg: the following would all match "text" or "test" : "te?t" , "t*t" , "te[sx]t" , "te[a-z]t").

To search for wildcard characters these need to be escaped with a forward slash.

eg: to search for "]" imap" you would use "\]" imap"

Also the highlight field has special formatting that allows you to specify any arbitrary valid HTML colour in curly brackets. If none is specified each highlighted term will be given a separate colour.

Further searching tips

The links at the bottom of the page are fully specified searches. Just click the link (after filling out any optional required information fields). After having the relevant log file lines you can get further information by clicking the message ID or thread ID several times. In the msg.log the first click will highlight entries, the second click will limit to these entries, the third click will attempt to display mail.log related to this message. In mail.log the first click will highlight entries related to this thread, the second click will limit to these entries and the third click will just center the search window on the item in question.

The highlight fields at the top of the log file output are dependant on what you are currently doing and will display some information that would be sensible to highlight based upon the log file you are currently searching.

Tracking a message

To track a particular message start with the link at the bottom of the page and drill down the queried links as much as you can.

eg Delivered and redirected to two local addresses

```
26 12:20:35 [1] Received 203.167.148.167 <fred@xxx> <user1@yyy> 366 <102@marijn> "In Queue"
26 12:20:35 [1] Stored 203.167.148.167 <fred@xxx> <redirect2@yyy> 366 <102@marijn> "Stored locally"
26 12:20:35 [1] Stored 203.167.148.167 <fred@xxx> <redirect1@yyy> 366 <102@marijn> "Stored locally"
```

eg. Delivered and retrieved via pop

```
26 12:46:38 [2] Received 203.167.148.167 <fred@xxx> <user1@yyy> 366 <103@marijn> "In Queue"
26 12:46:38 [2] Stored 203.167.148.167 <fred@xxx> <user1@yyy> 366 <103@marijn> "Stored locally"
26 12:47:29 [0] pop 127.0.0.1 "Fred" <fred@xxx> user1 394 <103@marijn> "103.marijn"
```

Format of msg.log

The msg.log files can be used to determine whether a particular message has been received and if it has been received what has happened to it. A new file is created on a daily basis named "msg<year><month><day>.rec". This log is a permanent record and is not rotated.

There are the following types of entries:

- Received - Message has been received via SMTP and queued for local or remote delivery
- Stored - Message has been delivered to local account
- Sent - Message has been delivered to remote account
- Later - Message delivery failed, will try again later
- pop - Message has been retrieved using POP3
- del - Message has been deleted using POP3 or IMAP

There are also the following entries

- New - Single line logged per recipient (requires g_log_rcpt)
- Fwd - Redirection has been applied (requires g_log_fwd)
- Rejected - Message has been received via SMTP and rejected (with reason)
- Tarpitted - Message has been tarpitted and rejected
- NoSubmit - Message was never submitted ([with reason](#))

Log entry format:

Entries for Received, Stored, Sent, Later have the same syntax:

```
"date & time" "queue id" Received "from ip" "from address" "dest address" "size" "id" "status"
```

```
"date & time" "queue id" tarpited "from ip" "from address" "dest address" 0 "status message"
"date & time" 0 pop "client ip" "from address" "pop login" "size" "xmailer id" "filename"
"date & time" 0 del "." "." "login" 0 "." "filename"
```

Format of mail.log

The mail.log - mail6.log contain information logged from within SurgeMail to record what is happening. There are three levels that are recorded error / information / debug. The information logging level can be set using the g_log_level setting.

This is the logfile NetWin support staff will probably request if you have any problems that need investigation.

Additional log files

- surgemail\startstop.log = Record of SurgeMail startup and shutdown
- webmail*.log (surgemail\scripts\webmail*.log) = Similar to mail*.log but for WebMail and must be explicitly enabled in wemail.ini
- surgemail\install.log = Logfile of what is done during installation
- surgemail\mon.log = Logfile of surgemail monitor (swatch.exe) activity
- surgemail\wweb.log = SurgeMail web serving activity
- surgemail\trace.log, mutex.log = Debug logs useful if SurgeMail dies unexpectedly

Particular questions

What does NoSubmit mean?

The "NoSubmit" log entry means that the message was never submitted. This could be for a variety of reasons such as an network problems, broken sending mailer, or surgemail filtering options.

```
24 01:07:39 [132061] NoSubmit 24.61.98.153 <support@domain.com> <evan@mydomain.net> 219
"tcp_read_dot"
```

```
It means the sending system sent
mail from: <yyy>
rcpt to: <xxx>
data
(closed connection)
```

In this case no 'message' was sent after the data command, this can occur if you have enabled g_badfrom_* checking because some systems take 20 seconds to respond to the from check and then their own sending stage timesout. You can add known domains that have this problem to the g_badfrom_whitelist.

eg. g_badfrom_whitelist "*"slow.domain,*other.domain"

```
24 01:07:39 [132061] NoSubmit 24.61.98.153 <support@domain.com> <evan@mydomain.net> 219
"No DATA command sent"
```

```
It means the sending system sent
mail from: <yyy>
rcpt to: <xxx>
(closed connection)
```

In this case no the connection was closed before the data command. This is likely to be the result of a g_badfrom check on an outgoing email by a system running surgemail with the g_bad_from check enabled. This is not a problem (NoSubmits was just not being logged previously).

Reporting

Three types of usage reports are available in surgemail, each provides slightly different information:

- Monthly usage reports
- Current status reports
- High use log file analysis

Monthly usage reports (fast to generate)

The Monthly usage reports are an efficient way of providing usage statistics on a per account, per domain or per server basis. This will typically be used for accounting purposes or for routine monitoring of the behaviour of users of the system.

A variety of statistics are recorded on a per account basis as each message is processed. These statistics can be broadly classified as relating to mail sent and received, and to mail retrieved via POP and IMAP. For local accounts both these sets of information are recorded. If surgemail is running in a surgewall configuration or has gateway rules defined, statistics are also gathered on the "gateway" accounts. These are obviously not real accounts on the system and this usage information is based on mail delivered to unique gateway email addresses in this case.

Sample report

Account usage (all domains) - matched 5 records today

account	msg Rcv	mb Rcv	msg Sent	mb sent	--	p/i sess	mb_read	--	~~~~~ wrapped below ~~~~~	
joebloggs@mydomain2.com	3	0.0	7	0.0	--	1	0.0	--		
marijn@mydomain.com	8	0.0	4	1.6	--	10	0.0	--		
user1@mydomain.com	4	1.6	0	0.0	--	1	0.0	--		
user1@mydomain2.com	4	0.4	0	0.0	--	0	0.0	--		
user2@mydomain2.com	14	1.4	0	0.0	--	0	0.0	--		
Total	33	3.4	11	1.6	--	12	0.0	--		

~~~~~

| last connected |          | lastip    | cpu_sec | elap_sec | mb quota | -- | virus blk | -- | msgsleft | mb_left |
|----------------|----------|-----------|---------|----------|----------|----|-----------|----|----------|---------|
| 12-Oct-2004    | 12:25:30 | 127.0.0.1 | 0       | 54       | 0.0      | -- | 0         | -- | 3        | 0.0     |
| 12-Oct-2004    | 12:20:09 | 127.0.0.1 | 0       | 14       | 0.0      | -- | 0         | -- | 0        | 0.0     |
| 12-Oct-2004    | 12:24:34 | 127.0.0.1 | 0       | 22       | 1.6      | -- | 0         | -- | 4        | 1.6     |
| never          |          |           | 0       | 0        | 0.4      | -- | 0         | -- | 0        | 0.0     |
| never          |          |           | 0       | 0        | 1.4      | -- | 0         | -- | 0        | 0.0     |
| never          |          |           | 0       | 90       | 3.4      | -- | 0         | -- | 7        | 1.6     |

~~~~~

The above report contains the following information:

- Number of messages received + size in Mb
- Number of messages sent + size in Mb
- POP / IMAP sessions + size of data read in Mb
- Time and IP address of last POP/IMAP connection
- Actual CPU time used (only accurate on windows) and elapsed CPU time
- Current quota used (in Mb)
- Viruses blocked on this account
- Messages and size of messages left online during last pop session

Domain usage reports provide a summary per domain with a total of all domains at the bottom. Account usage reports list each account individually with a total of all entries at the bottom. In addition the reports can be sorted on any column and limited to the top N accounts. In advanced mode wildcard exclude and limit filters can be applied similar to the main log file searching.

Available from

These reports are available to domain administrators and to server administrators using the Reports link in the web admin interface. In addition these reports are available on the command line using the following command: `tellmail report report_type [output file] [days=n sort_type=user ...]`

eg. Top 20 senders of mail over the last 5 days:

```
tellmail report account_usage c:\output.txt days=5 top=20 sort_type=bytesent
```

Valid values to pass each of the options of this tellmail command can be found by checking the na_reports.htm template file. You can pass the tellmail report command the following optional parameters: domain, days, sort_type, top, exclude, limit. In addition you can pass a date range over which to generate the report. To do so use "range=DDMMYYYY-DDMMYYYY" instead of "days=n".

Miscellaneous

The data for these reports is stored in daily usage snapshots (recYYMM/statYYMMDD.dat) with a monthly summary (recYYMM/summary.dat). When generating these reports it is most efficient to generate a report for "previous calendar month" as the monthly summary just gets loaded and appropriately filtered and displayed. In any of the n days to-date reports, each of the daily summary files gets totaled before the report gets generated. These statistics snapshot files are stored in binary format but can be dumped as text format using the command: tellmail pstat_dump filename

There are several other aspects one should be aware of as to the detailed operation of the mail sent and received logging:

- Under default installed operation "mail sent" is attributed to the account logged in using SMTP authentication. If the message is sent without SMTP authentication then the mail from address is used to determine the account sending mail. This can obviously be faked. The setting g_acctlog_authonly allows you to only record messages sent from a particular account if smtp authentication is used.
- For surgewalled accounts / gatewayed accounts the delivery of a message to a unique gatewayed email address will start the recording of mail sent and received on a particular account.
- When a message is redirected either from a particular account (eg user.cgi level forwarding or using aliases or redirection rules) this is attributed to a particular accounts as having "sent this message". This allows for the easy identification of accounts that have high volume redirections to non local addresses. This logging can be disabled using g_acctlog_noaliases.
- If you have a large number of inactive accounts in your userdatabase the recording of deliveries to these accounts can make the record files grow excessively large. If g_acctlog_sum_inactive is set, any mail delivered to an account that has never logged in yet is attributed to the not_logged_in_yet@yourdomain.com.

In addition to normal accounts usage information is also logged for the following special case accounts. Aliases as noted above, mailing lists and robots including the iss spam address. eg:

Account usage (all domains) - matched 5 records today

account	MSG Rcv	Mb Rcv	MSG Sent	Mb Sent	--	...
alias@mydomain.com	0	0.0	11	0.0	--	...
isspam@mydomain.com	0	0.0	12	0.0	--	...
testlist_marijn-bounce@mydomain.com	0	0.0	4	0.0	--	...
testlist_marijn@mydomain.com	1	0.0	0	0.0	--	...
c:\surgemail\some_robot.exe	4	0.0	0	0.0	--	...
isspam	70	0.5	0	0.0	--	...
Total	75	0.5	27	0.1	--	...

Current status reports (may be slow to generate)

The current status reports are generated based on actual users currently in the authentication database. If you have a slow authentication database and a large number of users the generation of this report can be rather slow.

Currently only a disk quota status report is available.

As for the monthly usage reports this report may be sorted on any column and limited to the top n entries or limited via the filters in advanced mode. Also this report is available to domain administrators.

Also as for the monthly reports this report can be generated on the command line using : tellmail report quota c:\output.txt top=20 sort=quota_pct

Account name	% used	bytes used	byte limit
user2@mydomain.com	34%	6.9mb	20.0mb
user1@mydomain.com	7%	1.5mb	20.0mb
marijn@mydomain.com	0%	8.0kb	20.0mb
user1@mydomain2.com	0%	392kb	unlimited
joebloggs@mydomain2.com	0%	2.6kb	unlimited
user2@mydomain2.com	0%	1.3mb	unlimited

Total NA 10.2mb unlimited

High use log file analysis (may be very slow to generate)

This report screen lets you analyse a section of the delivery logs on the fly to trouble shoot a particular "high use" problem.

- From IP address
- From Email address
- To Email address

The report can be sorted by the number of messages:

- Sent
- Stored
- Rejected
- Failed
- Spam
- Received

You can also specify the time period, and number of lines of the report to display. For Example:

Size	Stored	Rejected	Spam	Sent	Received	Failed	InQue	To
23k	0	1	0	0	2	0	1	chrisp@netwin.co.nz
4k	3	0	0	0	3	0	0	test3@lap
3k	0	1	0	0	0	0	0	spam2@netwin.co.nz

The complete delivery logs are processed each time for the generation of this report. If the log files are large then this is likely to be a rather disk io intensive process. For a typical system of 50K active users the delivery logs could easily be 250mb per day.

Managing Accounts

Account Creation

Account Creation - User Self Creation

Users can optionally create or sign-up for accounts via the web interface on the user port ie:

https://localhost:7443/cgi/user.cgi?cmd=user_check

There are several different methods, the method being used is specified per domain using the domain setting [create_user](#).

Account Creation - Manager

User administration is done via the "User accounts" option in the web admin contents.

<https://localhost:7025/cgi/admin.cgi>

This will display a page that allows you to lookup, create, modify and search for users. Once you have looked up an existing user you can change a user's password, edit any of the user's information including forwarding or delete them.

Domain administrators, configured by the [manager_username](#) setting, log in to the user self management interface:

<https://localhost:7443/cgi/user.cgi>

and have access to this users page for all the domains they manage.

Creating System Admin accounts

You can create additional system admin accounts for surgemail using the command:

```
surgemail -password
```

To remove accounts you will need to edit and manually remove entries from admin.dat

Interfacing with the user management system

The user management system is an HTTP based system using GET and POST commands to perform actions. If you want to provide some or all of these options elsewhere you can do so by interfacing with it using these HTTP commands. For example here is a simple HTTP form which will add a user.

However you should also consider the other two methods available to interface with accounts, which method you choose will depend on your precise needs.

- 1) the authent module can be talked to directly using the authent protocol <http://netwinsite.com/authent>
- 2) if the authent module is using a back end database (ldap, sql, etc...) then the backend database can be used directly.

To use the following examples to experiment with save them as htm files and open them in your browser. To use from a program you need to send the form as an http POST using tcpip. Many programming languages have a mechanism for easily sending web 'forms' from a program so will find this quite easy in some cases.

This group of forms provide simple access to the authent module functions directly, this can be used to add/del/modify/search user accounts. For these functions you must have the web admin user/password and send it using the normal http basic authentication method. While testing your web browser will prompt you for the user/password.

The action can be any of: nwauth_lookup, nwauth_set, nwauth_search, nwauth_del

The field 'a_info' is formatted like the raw authent module data, field="value" ... e.g.

```
fwd="user@domain.name" quota="20mb"
```

Here is a sample web page to test this with:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
```

```
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST" ACTION="http://localhost:7026/cgi/admin.cgi">
<p>
Action: <INPUT TYPE="text" NAME="cmd" VALUE="nwauth_lookup"> or nwauth_set nwauth_del nwauth_search<br>
Name <INPUT TYPE="text" NAME="a_user" VALUE=""><br>
Pass <INPUT TYPE="text" NAME="a_pass" VALUE=""><br>
Info <INPUT TYPE="text" NAME="a_info" VALUE=""><br>
<INPUT TYPE="submit" NAME="Submit" VALUE="Submit">
</FORM>
</BODY>
</HTML>
```

The following mechanisms use modified versions of the manager templates/interface to perform manager operations, for these a domain manager username/password is used.

A form to 'add' a user:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST"
ACTION="https://localhost:7443/cgi/user.cgi">
<br>Fields normally set to type Hidden to tell it what to do.
<INPUT TYPE="text" NAME="cmd" VALUE="cmd_user_login">
<INPUT TYPE="text" NAME="lcmd" VALUE="user_create">
<INPUT TYPE="text" NAME="show" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="user_fields" VALUE="user_id,quota,alias_quota,sms_quota,smsto">
<br>Domain admin user: <INPUT TYPE="text" NAME="username" VALUE="X"><br>
Domain Admin Pass: <INPUT TYPE="text" NAME="password" VALUE="Y"><br>
Name <INPUT TYPE="text" NAME="lusername" VALUE=""><br>
Pass <INPUT TYPE="text" NAME="lpassword" VALUE=""><br>
uid <INPUT TYPE="text" NAME="user_id" VALUE=""><br>
SMS Phone Number <INPUT TYPE="text" NAME="smsto" VALUE=""><br>
Disk Quota<INPUT TYPE="text" NAME="quota" VALUE=""><br>
Alias Quota<INPUT TYPE="text" NAME="alias_quota" VALUE=""><br>
SMS Quota<INPUT TYPE="text" NAME="sms_quota" VALUE=""><br>
<INPUT TYPE="submit" NAME="Submit" VALUE="Add user">
</FORM>
</BODY>
</HTML>
```

A form to 'delete' a user:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST" ACTION="https://localhost:7443/cgi/user.cgi">
<br>Fields normally set to type Hidden to tell it what to do.
<INPUT TYPE="text" NAME="cmd" VALUE="cmd_user_login">
<INPUT TYPE="text" NAME="lcmd" VALUE="user_delete">
<INPUT TYPE="text" NAME="show" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="user_fields" VALUE="user_id,quota,alias_quota,sms_quota,smsto">
<br>Domain admin user: <INPUT TYPE="text" NAME="username" VALUE="X"><br>
Domain Admin Pass: <INPUT TYPE="text" NAME="password" VALUE="Y"><br>
Name <INPUT TYPE="text" NAME="lusername" VALUE=""><br>
Pass <INPUT TYPE="text" NAME="lpassword" VALUE=""><br>
uid <INPUT TYPE="text" NAME="user_id" VALUE=""><br>
SMS Phone Number <INPUT TYPE="text" NAME="smsto" VALUE=""><br>
Disk Quota<INPUT TYPE="text" NAME="quota" VALUE=""><br>
Alias Quota<INPUT TYPE="text" NAME="alias_quota" VALUE=""><br>
SMS Quota<INPUT TYPE="text" NAME="sms_quota" VALUE=""><br>
<INPUT TYPE="submit" NAME="Submit" VALUE="Del User">
</FORM>
</BODY>
</HTML>
```

A form to Show user fields:

(as above but change lcmd)

```
<INPUT TYPE="text" NAME="lcmd" VALUE="user_details">
```

You may need to add the begin/end list to simple_msg.xml template e.g.

```
<?xml version="1.0" ?>
<rss version="2.0">
<xml_status>|xml_status|</xml_status>
<response>|message|</response>
<response2>|message2|</response2>
||begin_list||
<info_||user_info_label||>|user_info_value|</info_||user_info_label||>
||end_list||
</rss>
```

A form to Modify/Save changes to a user:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST" ACTION="https://localhost:7443/cgi/user.cgi">
<br>Fields normally set to type Hidden to tell it what to do.
<INPUT TYPE="text" NAME="cmd" VALUE="cmd_user_login">
<INPUT TYPE="text" NAME="lcmd" VALUE="user_update">
<INPUT TYPE="text" NAME="show" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="user_fields" VALUE="user_id,quota,alias_quota,sms_quota,smsto">
<br>Domain admin user: <INPUT TYPE="text" NAME="username" VALUE="X"><br>
Domain Admin Pass: <INPUT TYPE="text" NAME="password" VALUE="Y"><br>
Name <INPUT TYPE="text" NAME="lusername" VALUE=""><br>
Pass <INPUT TYPE="text" NAME="lpassword" VALUE=""><br>
uid <INPUT TYPE="text" NAME="user_id" VALUE=""><br>
SMS Phone Number <INPUT TYPE="text" NAME="smsto" VALUE=""><br>
Disk Quota<INPUT TYPE="text" NAME="quota" VALUE=""><br>
Alias Quota<INPUT TYPE="text" NAME="alias_quota" VALUE=""><br>
SMS Quota<INPUT TYPE="text" NAME="sms_quota" VALUE=""><br>
<INPUT TYPE="submit" NAME="Submit" VALUE="Modify User">
</FORM>
</BODY>
</HTML>
```

Example to create a domain (note this uses manager port):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST" ACTION="https://localhost:7025/cgi/admin.cgi">
<br> Fields normally set to type Hidden to tell it what to do.
<INPUT TYPE="text" NAME="cmd" VALUE="global_misc_save"> <br>
<INPUT TYPE="text" NAME="misc_settings"
VALUE="domain_name,url_host,manager_email,manager_username,manager_password,create_user,create_max">
<INPUT TYPE="text" NAME="misc_cmd" VALUE="special">
<INPUT TYPE="text" NAME="domainid" VALUE="-1">
<p>
(mx) Domain: <INPUT TYPE="text" NAME="name" VALUE="new.domain"><br>
(A) Host Name: <INPUT TYPE="text" NAME="url_host" VALUE="mail.new.domain"><br>
Manager email: <INPUT TYPE="text" NAME="manager_email" VALUE="test@new.domain"><br>
Manager account: <INPUT TYPE="text" NAME="manager_username" VALUE="test@new.domain"><br>
Manager password: <INPUT TYPE="text" NAME="manager_password" VALUE="secret"><br>
<p>
<INPUT TYPE="text" NAME="force_page" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="doframes" VALUE="false">
<INPUT TYPE="submit" NAME="Submit" VALUE="Add Domain">
</FORM>
</BODY>
</HTML>
```

Delete a domain:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST"
ACTION="https://localhost:7025/cgi/admin.cgi">
<INPUT TYPE="text" NAME="cmd" VALUE="domain_delete"> <br>
<INPUT TYPE="text" NAME="domainid" VALUE="-1">
<p>

Domain: <INPUT TYPE="text" NAME="domain" VALUE="new.domain"><br>
Delete users<INPUT TYPE="text" NAME="delete_users" VALUE="true"><br>
Delete files<INPUT TYPE="text" NAME="delete_files" VALUE="true"><br>
<p>
<INPUT TYPE="text" NAME="force_page" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="doframes" VALUE="false">
<INPUT TYPE="submit" NAME="Submit" VALUE="Delete Domain">
</FORM>
</BODY>
</HTML>
```

Check if a domain exists

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML><HEAD></HEAD><BODY><FORM NAME="Main" METHOD="POST"
ACTION="https://localhost:7025/cgi/admin.cgi">
<INPUT TYPE="text" NAME="cmd" VALUE="domain_exists"> <br>
<INPUT TYPE="text" NAME="domainid" VALUE="-1">
<p>

Domain: <INPUT TYPE="text" NAME="domain" VALUE="new.domain"><br>
Delete users<INPUT TYPE="text" NAME="delete_users" VALUE="true"><br>
Delete files<INPUT TYPE="text" NAME="delete_files" VALUE="true"><br>
<p>
<INPUT TYPE="text" NAME="force_page" VALUE="simple_msg.xml">
<INPUT TYPE="text" NAME="doframes" VALUE="false">
<INPUT TYPE="submit" NAME="Submit" VALUE="Check Domain">
</FORM>
```

```
</BODY>
</HTML>
```

For responses you can setup an xml file, something like this: simple_msg.xml contains (this is placed in the surgemail/web folder)

```
<?xml version="1.0" ?>
  <rss version="2.0">
    <xml_status>||xml_status||</xml_status>
    <response>||message||</response>
    <response2>||message2||</response2>
  </rss>
```

NOTE:

- The username and password fields, these should contain the username and password of a domain admin user (see [manager_username](#)).
- The user_fields field specifies the names of the fields containing user information to add.
- The new users name and password are supplied in the lusername and lpassword fields.
- The show field is used to select the output page. This file must exist in the surgemail web directory. It is not required to be XML, that was just used as an example of a format you might easily be able to parse. The content of the file may be absolutely anything you require.

Essentially what happens above is that SurgeMail carries out the cmd "cmd_user_login" using the username and password given. Once complete it looks for an lcmd, in this case user_create, it then copies all variables in the format "l<variable>" to "<variable>" so lusername is copied to username and lpassword is copied to password and then it executes user_create creating the new user.

This same process can be used with any lcmd value, provided you supply the fields required either directly (as is done with user_id, smsto, quota, alias_quota, ans sms_quota) or using l<variable> when they collide with variables required for the previous command i.e. lusername and lpassword above.

Because the show field allows you to control the format of the replies you can interface the user management system programatically, in other words write a complete custom user interface or add to an existing user interface. SurgeMail outputs a number of values which will aid in this, for example cmd_result which is equal to "success" or "failure". SurgeMail returns a [utoken](#) variable when a domain admin logs in, this utoken can be passed to carry out additional commands, instead of logging in for every request. Alternatively you can send requests to the [web admin interface](#), this method is often cleaner, easier and more flexible. To find out the values required for each command consider using [DSpy](#).

Using utoken

The above form essentially piggy backs a user_create command on a login command, required because you have to authenticate to create users. If you intend to send and receive the data using a program instead of a fixed html form then you have some better, and often simpler alternatives. The first is that you could send the login command by itself and store the resulting authentication token AKA "utoken". You can pass this utoken with another command instead of piggy backing every command on a login command.

Using the web admin instead

Another option uses the web admin interface instead of the domain admin one. You would connect to the web admin port 7026 (non https port) and send your request to /cgi/admin.cgi. This interface uses basic HTTP authentication, so instead of performing a login you simply need to send a special HTTP header with every request. This header contains the encoded username and password for the web admin interface, it's value is fixed, it does not change unless you change the username and/or password, so, find out the value and simply send that with every request. To find out the value consider using [DSpy](#).

Using DSpy

DSpy ([download](#)) is a simple windows application which listens on a specified port for connections. When it accepts a connection it also connects to another specified ip and port. It reads data from each connection and passes it all through to the other while logging the entire conversation to disk. This allows you to "spy" on the data going from one port to the other. It **cannot** be used with HTTPS ports to spy on encrypted conversations. For example you might tell it to listen on port 8080 and connect to 127.0.0.1 on port 7026, eg:

```
c:\>dspy -port 8080 -to 127.0.0.1:7026
```

Then you would type the following url into your browser:

```
http://127.0.0.1:8080/cgi/admin.cgi
```

and you should see the web admin interface. Perform the action you want your interface to be able to carry out, and look in the dspy.log file for the http headers and request data required to do it, duplicate the authentication header (if required) and the form content to perform this action with your own interface.

You can download dspy.zip [here](#).

If you need more information on this subject please contact surgemail-support@netwinsite.com.

Advanced Per User Services

Access to the SurgeMail's POP / IMAP / SMTP facilities can be controlled on a per user basis by defining a series of access groups ([g_access_group](#)). Users can belong to one or more access groups. Each access group has a IP based wildcarded limitation of POP, IMAP and SMTP access and is defined in surgemail.ini. The domain manager is able to change these settings for the user accounts within their domain.

Note that the groups defined by these settings can be used by many others settings to define those settings for users in these groups, eg. [g_user_access](#), [g_admin_access](#), [g_quota](#), [g_user_alias](#), [g_user_sms_quota](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_list_quota](#), [authent_info_grp](#), [web_access_grp](#)

Membership of this group is checked against the "Access Type" setting (NWAAuth field: mailaccess) in the authentication database.

eg. You could charge WebMail users for pop access privileges:

```
g_access_group group="paid_user" access_pop="*" access_imap="*" access_smtp="*"
g_access_group group="free_user" access_pop="webmail.svr.ip" access_imap="webmail.svr.ip"
access_smtp="webmail.svr.ip"
```

with "Access type" set to "free_user" / "paid_user" on accounts page or equivalently in NWAAuth authentication database:

```
marijn@mydomain.com:{sha}tVANQo...:created="1060034937" mailaccess="free_user" ...
```

Or the following would use three arbitrary fields (SuspendedEmail, AllowedPOP, AllowedIMAP) in the authentication database to define the whether POP / IMAP / Webmail / SMTP services can be used on an individual per user basis.

```
g_access_group group="webmailonly" access_pop="*" access_imap="1.2.3.4" access_smtp="*"
g_access_group group="imap" access_pop="*" access_imap="*" access_smtp="*"
g_access_group group="allowpop" access_pop="*" access_imap="*" access_smtp="*"
g_access_group group="suspendemail" access_pop="!*" access_imap="!*" access_smtp="!*"

g_group_field field="SuspendedEmail" value="0" group="webmailonly"
g_group_field field="SuspendedEmail" value="1" group="suspendemail"
g_group_field field="AllowedPOP" value="1" group="allowpop"
g_group_field field="AllowedIMAP" value="1" group="allowimap"

g_authent_info name="Suspended Email" field="SuspendedEmail" access="domadmin" default="0"
g_authent_info name="Allowed POP" field="AllowedPOP" access="domadmin"
g_authent_info name="Allowed IMAP" field="AllowedIMAP" access="domadmin"

marijn@mydomain.com:{sha}tVANQo...:created="1060034937" SuspendedEmail="0" AllowedPOP="*" AllowedIMAP="1"
```

In this case four groups are defined (webmailonly, SuspendedEmail, AllowedPOP, AllowedIMAP) with membership based on the (SuspendedEmail, AllowedPOP and AllowedIMAP) database fields. The use of g_group_field has the same effect as manually setting "Access Type" to the combined group membership of "webmailonly,allowimap" for user marijn@mydomain.com.

Actual access that is granted is worked out by processing the g_access_group rules in the order they are defined in surgemail.ini building up from no access. So if you want "suspendemail" to override "allowpop" you need to make sure the "suspendemail" g_access_group entry is after the "allowpop" g_access_group in the surgemail.ini file.

User access settings

There are many features provided in the user self management and domain administrator interfaces and it is common to want to enable or disable some or all of these features. SurgeMail uses user_access settings to do this, see [g_user_access](#) and [g_admin_access](#) for details on the various options.

Account Status

The account status field in the authentication database allows the domain controller to easily enable / disable individual mail accounts. This is the "Account Status" setting in the Web Admin interface (NWAAuth field: mailstatus). This setting can have one of the following values:

	Incoming SMTP	POP	WEB	IMAP	Sending SMTP	Response codes
ok,good	Y	Y	Y	Y	Y	
suspended	Y	N	N	N	N	Account suspended.
cancelled	N	N	N	N	N	Account cancelled
banned,bad	N	N	N	N	N	This account has been banned for inappropriate use.
closed	N	N	N	N	N	Account closed.
paydue,due	Y	Y	Y	Y	Y	
payup	Y	N	N	N	N	Please pay to continue service.
readonly	N	Y	Y	Y	N	Account is only allowed to read email.

These messages responses can be modified using the g_mailstatus_message:

```
g_mailstatus_message state="cancelled" message="Account cancelled,
please email postmaster@mydomain.com for further information"
```

or adding an "id" field to the authentication database and using this to supply a customised URL for updating account information:

```
g_mailstatus_message state="payup" message="Payment is overdue
for $full_name$, click http://myurl.com/cgi/mycgi.cgi?user=$id$ to update
account"
```

Individual user bounce message setting: bounce_msg

In versions 3.8g-12 and later you can use bounce_msg, e.g.

```
C:\surgeemail>nwauth -path .
lookup test2@star
+OK test2@star config 0 bounce_msg="please send to me@here.com"
```

Then when you send a message to that account you will get an smtp rejection:

```
550 Bounce_msg: please send to me@here.com
```

Restricting accounts to "local only"

You may want to setup a group of users (eg students, untrusted staff etc) who should be able to email within themselves and email trusted staff accounts but not email other internet accounts. There are several possibilities for implementing this in surgeemail. The easiest way is to implement this using g_user_send_rule / g_user_receive_rule rules:

1) Setup appropriate rules to allow inbound / outbound mail based on group membership and create an access group which you can use to apply these

```
g_user_receive_rule group="local" from="*@mydomain.com"
g_user_send_rule group="local" to="*@mydomain.com"
g_access_group group="local" access_pop="*" access_imap="*" access_smtp="*" access_incoming="*"
```

2) Assign membership of this group to the relevant accounts on the accounts page.

The rules above will allow accounts on mydomain.com that have "local" group membership to only send and receive mail to / from local accounts. Other possible relevant configuration options:

1) You could set this up as a separate "localonly" domain for easier management and configuration. In this case the rules are likely to be something like:

to allow sending between users and to main domain:

```
g_user_receive_rule group="local" from="*@localonly,*@mydomain.com"
g_user_send_rule group="local" to="*@localonly,*@mydomain.com"
```

to allow sending to main domain, but not between users:

```
g_user_receive_rule group="local" from="*@mydomain.com"
g_user_send_rule group="local" to="*@mydomain.com"
```

2) You may want to setup a restricted administrator account to allow user creation and addition and quota management but not change access group membership. In this case, make sure it setup as a separate localonly domain and then setup a rule something like:

```
g_authent_info_grp group="internal_manager" fields="none,quota" tag="dummy_tag"
```

and make sure the account to administer these restricted accounts is a domain manager, and also has assigned internal_manager group privileges.

3) A similar effect can be achieved with mfilter rules, but in this case full configuration is done in mfilter.rul and it has to be the main server administrator that makes any configuration changes.

eg:

```
recipients
  if (isin("from","fred@mydomain.com")) then
    if (!isin("recipient", "mydomain.com")) bounce "Sorry you can only send to mydomain.com"
  end if
end recipients
```


The tellmail command line utility

tellmail is a command line utility that allows you to perform some administrative tasks with SurgeMail such as adding users and domains to locating a users mailbox. To use tellmail you just need to type tellmail at the shell or command prompt. On Windows you can get to the command prompt by going left clicking on the start button, then program files, then accessories then command prompt.

Example:

C:\>tellmail <command>

All commands are preceeded by typing tellmail

General tellmail commands

[help](#)
[status](#)
[reload](#)
[shutdown](#)
[queue](#)

Misc. tellmail commands

[aspm_retrain](#)
[g_relay_allow_ip](#)
[suspend](#)
[resume](#)
[surgeplus](#)
[surgehost_update](#)

User commands

[path](#) <user@domain>
[add_user](#) <user@domain>
[delete_user](#) <user@domain>
[change_pass](#) <user@domain> <password>
[add_user_alias](#) <user> <alias>
[delete_user_alias](#) <user> <alias>
[add_domain](#) <domain>
[delete_domain](#) <domain>
[clear_cache](#) [user@domain]

[logout](#) <user@domain>
[showlocks](#)
[expire](#)
[expire_status](#)
[expire_user](#) <user@domain> <days> <bytes>
[set_authfield](#) fieldname file.txt [apply]
[find_user](#) <domain> <field_name> <field_value>
[user_send_max](#) <user@domain>
[add_member](#) <list> <user@domain>
[add_rules](#) <file>
[rescan_aliases_file](#)
[rescan_aliases](#)
[list_rcpt](#) <domain> [type]

Quota Commands

[quota](#) <user@domain> - Displays the users quota
[quota_rebuild](#) <user@domain> - Rebuilds user's quota.
[quota_domain](#) <domain> - Displays how much data the domain is currently using.
[quota_rebuild_domain](#) <domain> - Rebuilds the quota usage for this domain

The tellmail command line utility.

quota_set_domain <domain> <size> - Sets the quota for the specified domain
quota_resetall - Rebuilds all quotas for all users.
quota_set <user@domain> <amount> - sets the users quota.

Migration commands

[imap_import](#) <localuser@localdomain> <oldimaphost> <olduser> <oldpass> <delete|keep> <create|ncreate>

License commands

[activate](#)<registration> [email address]
[deactivate](#)<registration> [email address]

Mirroring

[resync_config](#)
[resync_fast](#)
[resync_nwauth](#)

help

Lists the commands you can use and a brief description

status

This gives you a ton of information on the server.

reload

This allows you to reload SurgeMail without having to stop and restart it which means it won't interrupt users that are currently online. Great if you have edited surgemail.ini manually and need to activate your changes without disrupting service.

shutdown

This simply tells SurgeMail to shutdown.

queue

This displays the current delivery queue.

path <user@domain>

This allows you to find where a users mailbox is stored on the disk.
example:

```
C:\>tellmail path stu@catch.netwin.co.nz  
C:\surgemail\mbox\catch.netwin.co.nz\xc\lg\stu\
```

add_user <user@domain>

This allows you to add users easily to SurgeMail.

delete_user <user@domain>

This allows you to delete users from SurgeMail. The deletion process is as follows.

- Deletes user from authentication database
- Removes cache entry
- Decreases registration count
- Decreases user count for domain
- Removes WebMail files (tells webmail to)
- Removes SurgeMail files (users mail)
- Removes aliases
- Removes mailing list subscriptions
- Records them in users_YYYYMMDD.rec

change_pass <user@domain> <pass>

This allows you to change users passwords easily.

add_user_alias <user> <alias>

This allows you to add aliases for users.

delete_user_alias <user> <alias>

This allows you to delete aliases for users.

add_domain <domain>

Adds a domain to SurgeMail. This command makes use of the domain_defaults.txt file specified [here](#).

delete_domain <domain>

Deletes a domain from SurgeMail.

clear_cache [user@domain]

This allows you to clear SurgeMail's authentication cache.

Examples:

tellmail clear_cache - With no argument it will clear the entire cache.

tellmail clear_cache test@localdomain - This will clear only this users cache out

tellmail clear_cache *@domain2.com - Clears the cache for all users at domain2.com

quota <user@domain>

Displays the users current disk usage and their allowed quota.

logout <user@domain>

This will logout a user that is currently connected to on IMAP or POP you can see the current locks by using the [tellmail showlocks](#) command

Example:

tellmail logout stu@blah.com

showlocks

This shows the current software locks used by IMAP and POP in SurgeMail. When a user logs into POP or IMAP SurgeMail creates a lock for that user to prevent multiple connections to the same account.

Example:

tellmail showlocks

Lock (catch.netwin.co.nz#stu#_) n=1 f=0 locktype=pop thid=2332

expire

This starts the expire process.

expire_status

Gives the status on the expire process.

expire_user <user@domain> <days> <bytes>

This expires mail in the specified users INBOX, and optionally Trash folders based on the specified criteria, you may specify an age in days and/or a max size in bytes, any message exceeding these will be expired.

find_user <domain> <field name > <field value >

The tellmail command line utility.

This lets you search for a user or users that match the criteria you have set.

Examples:

To get a list of accounts of a certain status:
tellmail find_user foo.com mailstatus suspended
tellmail find_user foo.com mailstatus payup
tellmail find_user foo.com mailstatus closed

To find out who is using friends
tellmail find_user foo.com friends true

To find users in a certain group
tellmail find_user foo.com mailaccess groupname

set_authfield fieldname file.txt [apply]

This allows you to apply new authent values for one field based on the text file file.txt and fieldname the name of the field to change. Without [apply] flag this is run in preview mode to see what changes to the database would be made and run with [apply] flag to actually make the changes.

where file.txt contains user, new value pairs
user1 @domain,value1
user2 @domain,value2
...

eg.
user1 @mydomain.com,20mb
user2 @mydomain.com,30mb

tellmail set_authfield quota newvalues.txt

Processing file (a.a) PREVIEWING CHANGES ...
2 records updated, 0 records did not exist
Original nwauth records stored in pre.txt and updated records in post.txt for review

user_send_max <user@domain>

Reports the number of emails the user has sent in the last 24 hour period.

add_member <list> <user@domain>

Adds the specified email to the specified mailing list.

add_rules <file>

Reads <file>, expects lines formatted:

email:rule

where "email" is the users email address i.e. user@domain.com and "rule" is a user exception rule formatted as expected in the users friend.rul file (create some rules for a user to discover the format, it can be different for different rules and we may change or extend it in future).

This command will not add duplicate rules. This command will only add rules for existing users.

rescan_aliases_files

scan aliases.txt, make domuser.newdat

rescan_aliases

scans domuser.dat, make aliases.txt

list_rcpt

Attempts to list all valid receipt addresses for the server. Can be run as:

tellmail list_rcpt

The tellmail command line utility.

```
tellmail list_rcpt domain.name  
tellmail list_rcpt domain.name [redirect|lists|alias|blog|misc]
```

The first gets all receipts.

The next gets only those for the specified domain.

The last gets only those for the specified domain and type.

**imap_import <localuser@localdomain> <oldimaphost> <olduser> <oldpass> <delete|keep>
<create|nocreate>**

You can use this command to import users from another server. This is for importing imap accounts.

1. The username and domain to which the account will be created in surgmail
2. The ip of the server you are migrating the account from
3. The login name of the account on the server you are migrating from.
4. The accounts password on the server you are migrating from
5. Whether you want to delete or keep the messages on the old server
6. Whether you want the account created on the surgmail server, unless you have already created the account this must be create.

Example:

```
old server: 10.0.0.5  
old user account: fred  
old account password: test  
username and domain to create on surgmail: fred@mydomain.com  
tellmail fred@mydomain.com 10.0.0.5 fred test keep create
```

activate <registration> [email address]

This tells SurgeMail to contact the netwinsite.com database and activate your SurgeMail server.

deactivate <registration> [email address]

This tells SurgeMail to deactivate SurgeMail. You should do this if you are going to move SurgeMail to a new machine and then can reactivate on the new machine.

resync_config

This tells SurgeMail to send the config file to the other mirror, you would run this on the machine that has the data you want transferred.

resync_fast

This tells SurgeMail to resend missing files to the slave, this uses a separate thread.

You run this on the machine that has the extra data, if both machines for some reason have data that each other doesn't have it is safe to run this command on both slave and master. This command only sends data the other machine does NOT have, so is considered a very safe command.

resync_nwauth

This resends NWAAuth to the other machine, usually run on the master, but if the master had a disk crash and lost NWAAuth you would run this on the slave.

resync_config

This resends the surgmail.ini to the other machine, usually run on the master, but if the master had a disk crash and lost it's config you would run this on the slave. This is **required** every time you enable g_mirror_config, i.e. add it, or go from FALSE to TRUE on either machine.

surgeplus

surgeplus is used for doing various SurgePlus commands. use "tellmail surgeplus" for a full list of SurgePlus commands. Some common surgeplus commands are

- **tellmail surgeplus status** - View current surgeplus status
- **tellmail surgeplus rebuild** - Delete cached client downloads so they will be rebuilt next time a user tries to download them. Useful if you have customized a SurgePlus image file.
- **tellmail surgeplus activate (24-digit-license-key)** - Tell SurgeMail to use your new SurgePlus license key.
- **tellmail surgeplus download** - Download the latest SurgePlus beta and release versions from netwinsite.com to make available to your users.
- **tellmail surgeplus reload** - Reloads and SurgePlus config files from disk.

aspam_retrain

You should only run this command if you manually change Asпам files.

g_relay_allow_ip <ip>

This allows you to check if an IP is allowed to relay or not.

eg tellmail g_relay_allow_ip 192.168.0.1

suspend

This command will stop SurgeMail from sending any mail out, instead it will just queue it until a tellmail resume command is issued.

resume

This command will make SurgeMail start sending the queue out again after it has been placed in suspend mode by the tellmail suspend command.

surgehost_update

This command will make SurgeMail generate a new surgehost.ini file for use with webmail. It will run through all the domains in surgemail and update surgehost.ini for each domain.

SurgeMail Configuration Settings Overview

This page is an automatically generated top level overview of all the surgemail settings.

Domain Specific Settings

1. [abook](#) - Define surgeweb shared address books for this domain
2. [access_group_default](#) - Default group to place users in
3. [admin_access_default](#) - Default admin features granted to domain admins in this domain
4. [alias_file](#) - Alias translation file for this domain, unix format
5. [alias_max](#) - Maximum number of aliases for this domain
6. [assume_created_epoch](#) - If user has no 'created' field assume they were created an arbitrarily large time in the past
7. [blogs_max_per_user](#) - Number of blogs each account can create
8. [broad_sync](#) - Broadsoft Sync Enable
9. [centipaid](#) - Enable CentiPaid feature for matching accounts
10. [class](#) - Define class of user for following commands to apply to
11. [comment](#) - Management notes and comments about the domain
12. [create_block](#) - Block new users from this ip
13. [create_cleanup](#) - Cleanup existing data before adding a user
14. [create_delete_days](#) - Number of days a disabled new account remains before deletion
15. [create_disable_days](#) - Number of days new accounts remain active for
16. [create_image](#) - Use verification image on signups
17. [create_linkto](#) - Link to redirect to after successful live account creation
18. [create_max](#) - Maximum signups from ip in time period
19. [create_repass](#) - Users must enter their password twice on creation
20. [create_reqd](#) - Required fields for new users, e.g. (phone,age)
21. [create_subdomain](#) - Allow users to have their own subdomain
22. [create_tpl_dir](#) - Relative directory (from /web) where 'netauth' pages are stored
23. [create_user](#) - Method for adding new users
24. [delete_user_after](#) - Number of days an account can remain unread before it is deleted
25. [disable_smtp_after](#) - Number of days an account can remain unread before delivery is disabled
26. [disable_surgeplus](#) - Disable SurgePlus Calendar/File Sharing
27. [dmail_bin_path](#) - Path for dmail bin files to automatically convert delivered mail
28. [dmail_deliver](#) - Deliver messages into dmail drop directories (not supported)
29. [dmail_drop_path](#) - Path for dmail / sendmail style drop files to automatically convert delivered mail
30. [dmail_drop_prefix](#) - Whether prefix is used on drop file names
31. [dmail_hash](#) - Hashing scheme used by dmail_drop_path and dmail_bin_path
32. [dmail_skip_imap](#) - Skip conversion of old imap *.mbx folders
33. [encrypt_limit](#) - Max encrypted msgs per user per hour
34. [encrypt_noconfirm](#) - Disable confirmation for encrypted messages
35. [encrypt_rule](#) - Matches will be encrypted when sent
36. [encrypt_smart](#) - Encrypt smart features enabled for this domain
37. [encrypt_subject](#) - Subject when encrypted message sent - default is original subject
38. [encrypt_token](#) - Send token to 'sender' for new SurgeVault recipients
39. [enotify_from](#) - From address to use in email notification messages
40. [expire_age](#) - Expire inbox mail older than (days)
41. [expire_rule](#) - Expire rules for specific folders, age is in days
42. [expire_size](#) - and larger than this
43. [fallback](#) - Default address for this domain, NOT RECOMMENDED
44. [fallback_always](#) - Also relay to old system even if user does exist - not recommended
45. [fallback_check](#) - Check if user exists on fallback_relay host before accepting it
46. [fallback_relay](#) - Host to send messages to if user doesn't exist here
47. [fallback_users](#) - Path to file listing all users to user fallback_relay for
48. [footer_file](#) - Text footer file for all messages 'from' this domain
49. [footer_html](#) - HTML footer file for all messages 'from' this domain
50. [forward_illegal](#) - Ban forwards to these addresses
51. [friends_at_rcpt](#) - Whether to check users friends list at rcpt stage

52. [friends_pending_name](#) - The imap name of the friends_pending folder default is 'Friends Pending'
53. [friends_url](#) - Specify full url for friends release http://domain.name:port domain specific setting
54. [from_exact](#) - Check from matches authenticated user
55. [gateway_to](#) - Send all email to another server
56. [header_add](#) - Add header to posts 'from' this domain
57. [host_alias](#) - Alternate name(s) for this domain
58. [imap_public](#) - Share IMAP folders between users
59. [imap_public_show](#) - Auto subscribe public folders
60. [language_default](#) - Default language for user web interface
61. [late_forward](#) - Apply domain users forwarding rules after friends, spam, and filtering
62. [ldap_anydomain](#) - Lets users search other than their own domain in ldap
63. [ldap_disable](#) - Stops ldap logins by users of this domain
64. [legal_archive_disable](#) - Disable legal archive for this domain (experimental)
65. [legal_archive_hide](#) - Hide legal archive for this domain (experimental)
66. [legal_archive_keep](#) - Days to keep legal archive, units=days unless you specify years or months
67. [list_disable](#) - Disables creation of mailing lists.
68. [list_max](#) - Maximum number of mailing lists for this domain.
69. [list_max_users](#) - Maximum number of users allowed in all lists in this domain.
70. [loginfails](#) - Disconnect user after this many password guesses
71. [lookup_relay_on_from](#) - Looks up local from addresses to check for relay="true"
72. [mailbox_path](#) - Path to mailbox (inbox) files
73. [manager_email](#) - Domain managers email address (for email based account creation confirmation)
74. [manager_username](#) - Domain managers username (for web based domain administration)
75. [msg_max_in](#) - Max size for messages to users in this domain, largest applied if multiple recipients
76. [msg_max_out](#) - Max size for messages sent by authenticated users of this domain
77. [old_imaphost](#) - Old IMAP server:port - transition IMAP accounts and folders if user doesn't exist
78. [old_imaphost_always](#) - Retrieve mail from old imap host on each login (slow - particularly for webmail)
79. [old_imaphost_createuser_disable](#) - Disable old_imaphost user creation on first login
80. [old_imaphost_file](#) - Migration based on file
81. [old_imaphost_lowercase](#) - Migration - All migrated folders are lowercase.
82. [old_imaphost_nodelete](#) - Leave mail on the old server (disables old_imaphost_always)
83. [old_imaphost_nodomain](#) - Strip domain from username when logging into old imap host
84. [old_imaphost_pass](#) - Migration based on file - password field
85. [old_imaphost_prefix](#) - Prefix for old imap server, e.g. mail//
86. [old_imaphost_skip](#) - Migration - Comma separate wild card list of migrate folders to skip past.
87. [old_imaphost_user](#) - Migration based on file - user field
88. [old_pophost](#) - Old pop server:port - transition accounts and pending messages if user doesn't exist
89. [old_pophost_always](#) - Retrieve mail from old pop host on each login
90. [old_pophost_bind](#) - Bind outgoing connection during pop migration
91. [old_pophost_createuser_disable](#) - Disable old_pophost user creation on first login
92. [old_pophost_nodelete](#) - Leave mail on the old server (disables old_pophost_always)
93. [old_pophost_nodomain](#) - Strip domain from username when logging into old pop host
94. [old_pophost_sep](#) - Separator, default is '@', e.g. some systems use %
95. [old_smtp_host](#) - SMTP host to check for existing users (when creating new accounts)
96. [old_smtp_host_skip](#) - Who to disable SMTP host checks for
97. [old_xfile](#) - Migration - Copy xfile data across
98. [pop_min_time](#) - Min seconds between pop logins (see warning)
99. [pop_welcome](#) - POP connection message
100. [prefix](#) - Database username prefix (deprecated, compatibility only)
101. [proxy_pop_nodomain](#) - Strip domain when talking to proxy pop host
102. [quota_default](#) - Default quota
103. [quota_domain](#) - Total quota for the domain, e.g. 300mb, 2gig
104. [rcpt_msg](#) - Response given for invalid recipient errors, message is prefixed by email address.
105. [redirect](#) - Redirect messages from 'was' to the new 'to' address
106. [redirect_cc](#) - Send carbon copy to another address
107. [redirect_hash](#) - Share incoming message evenly between several accounts
108. [redirect_max](#) - Limits the number of redirect rules
109. [security_suffix](#) - Suffix for smtp/imap/pop login
110. [send_helo](#) - Mail host A Record name used when sending helo to other servers - requires g_send_helo_from true

111. [smtp_auth_off](#) - Disable SMTP AUTH from unknown ip addresses
112. [smtp_from_ip](#) - Require incoming email from matching ip
113. [smtp_welcome](#) - SMTP connection message
114. [smtp_welcome_name](#) - SMTP welcome connection hostname
115. [spam_block](#) - Default for this domain to block spf etc failures
116. [spam_noblock](#) - Disable spf blocking for this domain
117. [spam_strip](#) - Strip spamdetect headers for this domain
118. [ssl_allow](#) - IP Wild card list to allow SSL encryption from
119. [ssl_pop_domain](#) - Domain to use for ssl certificates for POP and IMAP
120. [surgeplus_pop_server_name](#) - Default POP server for SurgePlus clients
121. [surgeplus_smtp_server_name](#) - Default SMTP server for SurgePlus clients
122. [surgeplus](#) - SurgeWall - Proxy this domain to specified mail server
123. [surgeplus_auth](#) - SurgeWall SMTP authentication
124. [surgeplus_capa_local](#) - Just return local imap capa response rather than remote
125. [surgeplus_local_too](#) - For web domain admin try local database too
126. [surgeplus_options](#) - Various SurgeWall options
127. [surgeplus_backend_server](#) - Backend server to connect to
128. [surgeplus_backend_smtp](#) - Backend smtp access (if non default)
129. [surgeplus_backend_web](#) - Backend web access - for usercgi /surgeplus (if non default)
130. [surgeplus_custom](#) - Surgeplus customisation level
131. [suspend](#) - Disable logins for entire domain
132. [url_alias](#) - Allows translation from one url to another
133. [url_blogs](#) - BLOGS host A Record name (if different from MX Record name - eg. blogs.mydomain.com)
134. [url_host](#) - Mail host A Record name (if different from MX Record name - eg. mail.mydomain.com)
135. [user_access_default](#) - Default user features granted to users in this domain
136. [user_alias](#) - Number of aliases accounts can create
137. [user_auto](#) - Auto create users when a login attempt occurs
138. [user_auto_pass](#) - Auto create users with this password on message delivery
139. [user_centipaid](#) - User Centipaid configuration options
140. [user_list_quota](#) - Number of mailing lists users can create
141. [user_max](#) - Maximum number of users in this domain
142. [user_send_max](#) - Maximum number of emails per day (requires SMTP AUTH)
143. [user_sms](#) - Allow users to set up sms notifications
144. [user_sms_quota](#) - Number of sms messages per account
145. [user_status_send](#) - How often to send user status messages (0 = never)
146. [web_access_ip](#) - Restrict access to web ports based on ip
147. [web_path](#) - Path to web admin pages
148. [web_url_path](#) - Url to path translation with access specifier
149. [webdav_quota](#) - Webdav quota per user in this domain, e.g. 100mb
150. [webmail_host](#) - The ip address or name of the machine to instruct webmail to connect to.
151. [webmail_url](#) - Url to the WebMail cgi
152. [webmail_urladd](#) - Url data to append to WebMail auto-login link
153. [webmail_workarea](#) - Path to WebMail workarea
154. [xfile_url](#) - Url to xfile files (see surgeplus utility)

Global settings

1. [g_access_group](#) - Grouped per user access limitations
2. [g_access_group_default](#) - Default group to place users in
3. [g_acctlog_authonly](#) - Log sending usage based on authenticated accounts only and ignore "MAIL FROM" address - which may be fake!!
4. [g_acctlog_noaliases](#) - Don't log redirection & aliases as sending mail as a result of redirection / forwarding (means you will not log account forwarding usage)
5. [g_acctlog_sum_inactive](#) - Summarise local accounts that have not logged in yet as not_loggedin_yet@domain.com
6. [g_admin_access](#) - Domain admin features granted to access groups
7. [g_admin_access_default](#) - Default admin features granted to domain admins
8. [g_admin_guesses](#) - Max guesses per IP for web admin access, e.g. 15
9. [g_admin_ip](#) - Mask of valid IP addresses for web admin users (default *)

10. [g_admin localhost](#) - Allow localhost web admin without user/pass
11. [g_admin utoken expire](#) - Length of time a web admin session is valid for
12. [g_admin utoken idle](#) - Length of time a web admin session may remain idle for
13. [g_alias login disable](#) - Disable user login as alias
14. [g_allow bodyless](#) - If true bodyless mail messages will be accepted (usually spam)
15. [g_allow passzip from](#) - A list of addresses to allow unmonitorable archive messages to be sent from
16. [g_allow passzip to](#) - A list of addresses to allow unmonitorable archive messages to be sent to
17. [g_allow_user_authent_field_get](#) - A space separated list of authent process fields that users are allowed to view for themselves using the POP xauthent_field_get command
18. [g_allow_user_authent_field_set](#) - A space separated list of authent process fields that users are allowed to set for themselves using the POP xauthent_field_set command
19. [g_archive](#) - Archive messages that match these rules
20. [g_archive bucketsize](#) - Size for archive bucket files. Default is 1mb
21. [g_archive early](#) - If true apply archiving before filtering is applied (superceeded by early flag on g_archive)
22. [g_archive files](#) - Archive attachments to a directory
23. [g_archive on delete](#) - Don't delete user files, archive them to g_archive_on_delete_dir
24. [g_archive on delete dir](#) - Directory to archive user files to on delete
25. [g_archive tcpip](#) - Rules for TCPIP archive process
26. [g_archive tcpip host](#) - Host to send archive data too
27. [g_aspam headers](#) - Add aspam information messages to messages.
28. [g_aspam need ip](#) - Require good matches to match external ip address
29. [g_assume_created epoch](#) - If user has no 'created' field assume they were created an arbitrarily large time in the past
30. [g_atrn client](#) - Define a rule for fetching email using ATRN protocol
31. [g_atrn port](#) - Port to listen for 'atrn' (On Demand Relay) requests
32. [g_atrn server](#) - On Demand Mail Relay settings to define user/pass for clients to fetch mail
33. [g_auth hide](#) - Disable SMTP Authentication for this IP List/Wild card address
34. [g_auth norelay](#) - Ignore SMTP auth for relaying purposes
35. [g_auth path](#) - Path to nauth files
36. [g_auth skipgateway](#) - Skip gateway rules if we get a proxy SMTP auth command
37. [g_authent allow badascii](#) - Allow ascii chars outside the range 32 < 127
38. [g_authent always](#) - Always lookup user, so virtual domains can exist just in authent module, loses existing users files
39. [g_authent any](#) - Restore buggy behaviour of looking up users in domains that don't exist
40. [g_authent cachebad](#) - Set the life in seconds that the cached failed lookups can be used, default 60 seconds
41. [g_authent cachelife](#) - Set the life in seconds that cached authent lookups can be used, default 1 hour
42. [g_authent cachesize](#) - Set the size of the authent cache, default is 500 entries
43. [g_authent case sensitive](#) - Make passwords case sensitive
44. [g_authent decrypt](#) - Collect and store plain text passwords for migration in file pass.decrypted
45. [g_authent domain](#) - If true add @virtual.domain.name to external user lookups, replaced with g_authent_nodomain setting
46. [g_authent encrypt key](#) - Encryption key for ccnumber auth field
47. [g_authent fwdfile](#) - Enables reading of old dmail .fwd files
48. [g_authent info](#) - User info names, fields and access rules
49. [g_authent info grp](#) - Fields to show to users in this group
50. [g_authent ip](#) - Lookup ip numbers in authent database with @ip added, to find send_limit=n values, must define tarpit_max_remote and g_tarpit_drop
51. [g_authent last login](#) - Store users last login time in the database
52. [g_authent logall](#) - Turns on logging of authent requests
53. [g_authent nodomain](#) - If true dont add @virtual.domain.name to external user lookups (NOT RECOMMENDED)
54. [g_authent number](#) - Number of authent processes to run
55. [g_authent path broken](#) - Allow authent module to return drop path, strongly discouraged, and BROKEN!!
56. [g_authent prefix sep](#) - Prefix separator, defaults to an underscore, a single character
57. [g_authent process](#) - Authent process command line
58. [g_authent reminders](#) - Days till we remind user to change password
59. [g_authent restart](#) - Cycle auth modules every 1000 lookups
60. [g_authent single](#) - Allow local users with a single quote char in their name
61. [g_authent spaces](#) - Allow spaces in passwords DO NOT USE
62. [g_authent strip domain](#) - Domain to strip when doing authent lookups

63. [g_authent_timeout](#) - Timeout for authentic response, default 60 seconds
64. [g_autologin_file](#) - File to use to share auto login information on NFS based cluster
65. [g_autologin_imap_disable](#) - Disable IMAP based autologins
66. [g_autologin_pop](#) - Performs auto-logins using pop3, used by webmail
67. [g_backtrace_disable](#) - If true backtrace code is disabled on unix
68. [g_bad_login_allow](#) - Number of consecutive bad logins for a user before blocking that user
69. [g_bad_login_ip_allow](#) - Number of bad logins from an ip before blocking that ip
70. [g_bad_login_ip_ignore](#) - IP address(es) to ignore bad logins from
71. [g_bad_login_mins](#) - Minutes to block login for, if consecutive bad ones received
72. [g_badfrom_badmx](#) - If mx host is one of these addresses then drop the message, it's definitely spam (e.g. 127.*)
73. [g_badfrom_check](#) - Check env from by connecting to it, always tick 'stamp' rule too or messages will bounce! NOT RECOMMENDED
74. [g_badfrom_from](#) - From to use when doing the check, not normally needed, if set must be set to valid account
75. [g_badfrom_noip](#) - Check envelope from domain exists and is a valid ip number, if not reject message
76. [g_badfrom_noip_temp](#) - Makes g_badfrom_noip return a temporary error instead of a 501 error
77. [g_badfrom_stamp](#) - Instead of bouncing message, just stamp a header to show if from address is no good
78. [g_badfrom_whitelist](#) - List of domains that we don't try badfrom checking on (see g_smite_skip)
79. [g_ban_blackhole](#) - Leave connected but reject all recipients without looking them up
80. [g_ban_from](#) - Disconnect if this wild card matches the from envelope
81. [g_ban_helo](#) - Disconnect if user says 'helo xxx' (or wildcard)
82. [g_ban_rcpt](#) - Disconnect any user delivering to this address/wildcard
83. [g_bank_debug](#) - Log request to bank server
84. [g_bank_group](#) - Create price groups with descriptions
85. [g_bank_log](#) - Log lines matching this in response.
86. [g_bank_ok](#) - Find this in response, if found then charge was successful
87. [g_bank_pass](#) - Password for authenticated web request to banks system
88. [g_bank_reason](#) - This line is returned to user if it is found
89. [g_bank_url](#) - URL to charge a credit card (experimental)
90. [g_bank_user](#) - Username for authenticated web request to banks system
91. [g_bind_authent_default](#) - Bind to default if authenticated
92. [g_bind_byfromip](#) - Bind outgoing SMTP connections to the specified IP based on the sender IP
93. [g_bind_from](#) - Bind outgoing SMTP connections based on 'from' envelope
94. [g_bind_incoming](#) - Bind outgoing SMTP connections based on incoming ip address
95. [g_bind_out](#) - Bind outgoing SMTP connections to this IP
96. [g_black_above](#) - Level for spam detection for blacklisting ip number e.g. 7
97. [g_black_count](#) - Number of spam in a row before we blacklist ip for 30 minutes, e.g. 30
98. [g_black_isspam](#) - Blacklist ip address for any spam training event
99. [g_black_nbad](#) - Blacklist ip address if this many bad recipients in a row (e.g. 8)
100. [g_black_to](#) - Blacklist ip address if they deliver to these user@domain addresses
101. [g_black_white](#) - Whitelist to prevent blacklisting, e.g. 1.2.3.*,mail*.aol.com
102. [g_block_files](#) - Wild card list of files to bounce, e.g. *.exe,*.cmd
103. [g_block_longok](#) - If true allow long file names (more than 180 char)
104. [g_block_skip](#) - From or To address to bypass g_block_files
105. [g_block_wild](#) - Block wild cards in usernames
106. [g_blogs_allow_links](#) - Allow users to post comments that contain urls
107. [g_blogs_cleanup_links](#) - Delete existing posts that contain urls
108. [g_blogs_comment_rev](#) - Show blog comments newest first
109. [g_blogs_default_template](#) - Default template set that is used by newly created blogs
110. [g_blogs_donly](#) - Only list blogs in a users domain
111. [g_blogs_enable](#) - Surgemail blogs
112. [g_blogs_image_optional](#) - Allow users to specify if image verification is required for comments
113. [g_blogs_max_per_user](#) - Maximum number of blogs per user
114. [g_blogs_maximum_image_size](#) - Default maximum image size
115. [g_blogs_maximum_image_width](#) - Default maximum image width
116. [g_blogs_maximum_items_in_top_page](#) - Maximum number of items on the top blog page
117. [g_blogs_no_suffix](#) - Shortens URL, url_blogs must be defined for each domain
118. [g_blogs_not_global](#) - Only allows access to a blog on the domain it is defined on
119. [g_blogs_not_unique](#) - Allow the same blog name in multiple domains

120. [g_blogs_ping](#) - Sites to ping on each post
121. [g_blogs_sub_domain_prefix](#) - Prefix to use instead of blogs. for blog subdomains. use ! to have no prefix.
122. [g_blogs_use_sub_domains](#) - Make blogs accessible at http://blog_name.domain/
123. [g_body_filter](#) - Enable user email body filtering
124. [g_bomb_max](#) - Max msgs to a single email address/hour
125. [g_bomb_max_from](#) - Max msgs from a single email address/hour
126. [g_bomb_white](#) - don't apply bomb_max limit if to address matches
127. [g_bounce_bind](#) - Use a specific ip address for outgoing bounces
128. [g_bounce_disable](#) - Disable all bounces (NOT A GOOD IDEA)
129. [g_bounce_limit](#) - Max size in bytes of message to send back as bounce, message is truncated if necessary
130. [g_bounce_nodrop](#) - Enables locally generated bounces for non local users
131. [g_bounce_paranoid](#) - Prevent external bounces going through surgemail
132. [g_bounce_redirect](#) - Send all bounces to a local address
133. [g_bounce_reject](#) - Reject bounces by ip address from known dumb mail servers
134. [g_bounce_some_stop](#) - Disables locally generated bounces for partial message failure - NEVER use this!
135. [g_bounce_suggest](#) - Send bounces to postmaster if spf cannot be verified
136. [g_bounce_to](#) - Domains to treat as local and send bounces to
137. [g_bounce_to_recipient](#) - Bounce suregwall failure to the recipient
138. [g_breakin_enable](#) - Stop multiple ip logins for one account in a few seconds
139. [g_breakin_white](#) - Email addresses that can send from multiple ips (or * to disable this feature)
140. [g_broad_pass](#) - BroadSoft pass
141. [g_broad_port](#) - BroadSoft port
142. [g_broad_server](#) - URL to BroadSoft server
143. [g_broad_url](#) - URL to this server
144. [g_broad_user](#) - BroadSoft user
145. [g_bull_rule](#) - Post bulletins to this domain
146. [g_centipaid](#) - CentiPaid address and port
147. [g_cid_skip_to](#) - Skip CID score, good for lawyers etc
148. [g_comment](#) - Management notes and comments about the server
149. [g_con_perip](#) - Connection limit per ip - sum of SMTP/POP/IMAP (if over refuse connection)
150. [g_con_perip_except](#) - Exception IP addresses to g_con_perip
151. [g_con_persubnet](#) - Global concurrent connection limit per ip subnet x.x.x.*
152. [g_convert_percent](#) - Convert % to @ sign in rcpt address
153. [g_country_ip](#) - Tag messages with country of origin
154. [g_crash_normal](#) - Crash without catching signals 10,11 so good core dump on freebsd
155. [g_create_allow](#) - List of characters allowed in usernames (and passwords, unless g_create_allow_pass is used)
156. [g_create_allow_pass](#) - List of characters allowed in passwords
157. [g_create_apply](#) - List of user groups to apply create_* settings for.
158. [g_create_badnames](#) - List of illegal usernames
159. [g_create_cleanup](#) - Cleanup existing data before adding a user
160. [g_create_dictionary](#) - File containing dictionary words to compare passwords to
161. [g_create_pass_digit](#) - Require one digit and letter in a password
162. [g_create_pass_length](#) - Limit the length of user passwords
163. [g_create_pass_mixed](#) - Require mixed case passwords
164. [g_create_pass_slack](#) - Slacken restrictions on trivial password creation
165. [g_create_record_ip](#) - Causes surgemail to store ipnum in the authentic database
166. [g_create_strict](#) - Whether to apply strict rules to usernames/passwords
167. [g_create_strict_admin](#) - Enforce strict rules for admins too, set g_create_strict AS WELL!!
168. [g_create_user_length](#) - Limit the length of usernames
169. [g_dbabble_links](#) - Add web links to DBabble from other web interfaces (and vice versa)
170. [g_dbabble_smtp_port](#) - DBabble SMTP port (do not manually change this setting - it should be set from the DBabble section of the web admin interface only)
171. [g_dbabble_smtp_prefix](#) - DBabble SMTP prefix (do not manually change this setting - it should be set from the DBabble section of the web admin interface only)
172. [g_debug_block](#) - For catching bugs in block file processing
173. [g_debug_crt](#) - Some CRT debugging on windows, do not use
174. [g_debug_free](#) - Check free memory isn't corrupted - slows performance slightly
175. [g_debug_imap](#) - Log imap folder renames and deletes in kmsg.log
176. [g_debug_ini](#) - Debugging, don't use this

177. [g_debug_vanished](#) - Name of file to check for, if file vanishes, crash
178. [g_delete_exclude](#) - Field and value that excludes an account from g_delete_user_after
179. [g_delete_user_after](#) - Number of days an account can remain unread before it is deleted (SEE WARNING).
180. [g_delete_user_mode](#) - Action when account is due to delete (write a command file etc...)
181. [g_delete_user_suspend](#) - If suspending an unread account set this field/value.
182. [g_deny](#) - Block users from some ip addresses
183. [g_deny_msg](#) - Block users from some domains
184. [g_deny_smtp](#) - Block users from some ip ranges connecting to SMTP only
185. [g_disable_exclude](#) - Field and value that excludes an account from g_disable_smtp_after
186. [g_disable_skip](#) - Ip address of senders to accept email from even if user account is disabled due to g_disable_smtp_after
187. [g_disable_smtp_after](#) - Number of days an account can remain unread before delivery is disabled
188. [g_disable_surgeplus](#) - Disable SurgePlus Calendar/File Sharing
189. [g_disable_surgeplus_updates](#) - Disable automated downloading of new versions of SurgePlus client from netwinsite.com
190. [g_diskio_abort](#) - Shutdown if diskIO failure on queue files
191. [g_dkim_check](#) - DKIM Check incoming DKIM signatures
192. [g_dkim_exclude](#) - DKIM Domains to not sign for outgoing email
193. [g_dkim_headers](#) - DKIM List which headers to sign (blank=default, and is usually best)
194. [g_dkim_only](#) - DKIM Domains to sign for outgoing email (default is all)
195. [g_dkim_selector](#) - DKIM Policy name for your server (used creating dns entry for dkim)
196. [g_dkim_sign](#) - DKIM Sign outgoing messages
197. [g_dkim_skip](#) - DKIM Desetination Domains to not sign
198. [g_dlist_nolocal](#) - Remove add local button from mailing lists
199. [g_dlist_nostart](#) - If set then don't start dlist (dmail compatibility)
200. [g_dlist_path](#) - DList Path normally defaults to \$g_home/dlist
201. [g_dmail_filter](#) - DMail compatible filter.txt file
202. [g_dns_cache_size](#) - Set size of forward dns cache, default 7000
203. [g_dns_host](#) - Hosts to send DNS lookups to
204. [g_dns_match_msg](#) - Message for stamp or bounce if forward and reverse lookup don't match
205. [g_dns_nlookup](#) - Concurrent DNS lookups to send to DNS server, default=20
206. [g_dns_nocache](#) - Disables DNS cache for spf lookups (20 minute life)
207. [g_dns_nopr](#) - Set to reject or retry, for ip addresses with no reverse dns entry (rdns)
208. [g_dns_nopr_msg](#) - Message for stamp or bounce if DNS lookup fails on ip address
209. [g_dns_nopr_skip](#) - Skip RDNS for these ip addresses
210. [g_dns_paranoid](#) - Compare forward and reverse dns lookup and check they match (set to STAMP or REJECT) not recommended
211. [g_dns_require](#) - Require MAIL FROM header matches senders ip reverse dns
212. [g_dns_system](#) - Use system code to do reverse lookups
213. [g_dns_translate](#) - If mx response is x.x.x.x translate to y.y.y.y:port
214. [g_domadmin_utoke_expire](#) - Length of time a domain admin login token is valid for in seconds
215. [g_domadmin_utoke_idle](#) - Length of time a domain admin login token may remain idle for
216. [g_domain_create_auto](#) - Auto create domain if it doesn't exist when creating a user
217. [g_domain_create_route](#) - Auto create route to mx mail server
218. [g_domain_default](#) - Default domain if user does not enter a domain on pop/imap login
219. [g_domain_list_max](#) - Maximum number of domains to list at once
220. [g_domain_separator](#) - Separator character for virtual domains
221. [g_domain_templates](#) - Check for domain specific templates
222. [g_domainkeys_check](#) - Check incoming DomainKeys signatures (obsolete turn off)
223. [g_domainkeys_headers](#) - List which headers to sign
224. [g_domainkeys_only](#) - Domains to sign for outgoing email
225. [g_domainkeys_selector](#) - Policy name for your server (used creating dns entry for domainkeys)
226. [g_domainkeys_sign](#) - Sign outgoing messages (obsolete, turn off)
227. [g_domuser_file](#) - Domain user file. Create thousands of virtual domains easily
228. [g_dotlock_minutes](#) - Minutes to wait for NFS lock file, default 20 minutes
229. [g_dotstuff_fix](#) - Debugging setting, do not change or bad things will happen
230. [g_doweb](#) - Do web part only
231. [g_download](#) - Fetch an http file and do an ini reload
232. [g_drop_use_len](#) - Use the content-len header for drop file processing (Solaris)
233. [g_dsn_enable](#) - Enable DSN (Delivery Status Notification) esmtp extension.

- 234. [g_dsn_nofinal](#) - Try not to show real final recipients but just original recipients
- 235. [g_ehlo_8bitmime](#) - Enable 8bit mime in ehlo response (not recommended)
- 236. [g_ehlo_fault](#) - Internal - for generating/testing faulty ehlo responses
- 237. [g_ehlo_simple](#) - Ip addresses to give simple ehlo response to
- 238. [g_emailreg_enable](#) - Enable whitelist <http://www.emailreg.org> register to use
- 239. [g_encrypt_config](#) - Encrypt some config settings (passwords)
- 240. [g_encrypt_disable](#) - Disable encryption
- 241. [g_encrypt_expire](#) - Days to keep encrypted messages, default 60
- 242. [g_encrypt_inline](#) - Use INLINE method by default
- 243. [g_encrypt_limit](#) - Max encrypted msgs per user per hour
- 244. [g_encrypt_max](#) - Max encrypted per day server wide
- 245. [g_encrypt_nodomain](#) - Allow encryption for users without local domains
- 246. [g_encrypt_none](#) - Don't encrypt if subject starts with this
- 247. [g_encrypt_path](#) - Path to encrypted files, this is not supported when mirroring!
- 248. [g_encrypt_prefix](#) - Prefix for encrypted messages must match encrypt rule so replies are encrypted
- 249. [g_encrypt_pw_host](#) - Central host for encryption password storage
- 250. [g_encrypt_pw_key](#) - Central host password key
- 251. [g_encrypt_reminders](#) - Days before we send users a reminder to change passwords, not recommended
- 252. [g_encrypt_reply_plain](#) - Send plain message for local replies
- 253. [g_encrypt_reset_msg](#) - Msg Body sent when password has been reset
- 254. [g_encrypt_reset_safe](#) - When users password is reset, delete all messages to them
- 255. [g_encrypt_smart](#) - Smart Encrypt Private Feature (not available)
- 256. [g_encrypt_ssl_force](#) - Require ssl on incoming encrypted messages
- 257. [g_encrypt_ssl_noforce](#) - Exceptions, e.g. surgeweb or localhost
- 258. [g_encrypt_surgeweb_show](#) - Show SurgeVault in SurgeWeb
- 259. [g_encrypt_unlock](#) - Unlock for these destinations. e.g. user@domain
- 260. [g_enotify_from](#) - From address to use in email notification messages
- 261. [g_eof_fix_off](#) - Turns off auto stripping of control+Z
- 262. [g_error_xlate](#) - Change error messages
- 263. [g_event_list](#) - Events wanted by url
- 264. [g_event_url](#) - Send msg events to a url
- 265. [g_expire_every](#) - Only expire spool once every 'n' days
- 266. [g_expire_silent](#) - Don't send users emails telling them what was expired.
- 267. [g_expire_trash](#) - Expire any messages found in trash folders
- 268. [g_expire_warning](#) - Give warning 'n' days before deleting each file
- 269. [g_external_ip_disable](#) - Do not add X-External-IP header
- 270. [g_fallback](#) - Default address for all local domains
- 271. [g_fallback_relay_if_exists](#) - Use FALLBACK_RELAY if not logged in but user exists
(OLD_POPHOST_CREATEUSER_DISABLE)
- 272. [g_feat_testing](#) - Testing setting do not use
- 273. [g_filter_max](#) - Max size for items to be sent to filter_pipe, or g_user_pipe, default no limit
- 274. [g_filter_n](#) - Concurrent filters to run at one time, default is 20
- 275. [g_filter_pipe](#) - Filter program that accepts msg on stdin and sends on stdout
- 276. [g_filter_pipe_noauth](#) - Skip for auth users
- 277. [g_filter_pipe_skip](#) - Skip filter if ip matches this
- 278. [g_filter_timeout](#) - Filter timeout in seconds, default is 360
- 279. [g_fix_crcrlf](#) - Fix email messages containing crcrlf for line termination
- 280. [g_fix_imap_if](#) - During IMAP import fix email messages containing If
- 281. [g_footer_auth](#) - Only add footer for authenticated local users
- 282. [g_footer_file](#) - Footer file which is appended to all mail messages
- 283. [g_footer_html](#) - HTML Footer file which is appended to all mail messages
- 284. [g_footer_notfound](#) - Only add footer if footer is not in message already
- 285. [g_footer_send](#) - Footer file added to outgoing messages only
- 286. [g_footer_sendonly](#) - If true only add footers when sending to non local users
- 287. [g_footer_skip](#) - Skip footers for these users
- 288. [g_footer_trusted](#) - Only add footers if sender is trusted
- 289. [g_forward_attach](#) - When late forwarding send as attachment to these domains
- 290. [g_forward_illegal](#) - Ban forwards to these addresses
- 291. [g_forward_oops](#) - Internal testing setting, not for general use sorry
- 292. [g_friends_add_trusted](#) - Add to friends list when if sender is trusted

293. [g_friends_allow_spf](#) - Allow all email through as if it was a friend during temporary allow
294. [g_friends_always](#) - Always use friends list.
295. [g_friends_at_rcpt](#) - Whether to check users friends list at rcpt stage
296. [g_friends_bounce_friend](#) - Allow exception rules to bounce a mesesage from a friend
297. [g_friends_bounce_rej](#) - Reject blank return path as friends failures
298. [g_friends_bounce_second](#) - Bounce the next time the user sends a message if waiting for confirm still
299. [g_friends_byemail](#) - Use old email based friends rejections
300. [g_friends_check_spf](#) - Disable friends bounces if SPF headers missing/failed to avoid backscatter.
301. [g_friends_confirm_debug](#) - Log sucessful friends confirmation responses
302. [g_friends_confirm_subject](#) - String to use as the subject of a friends confirmation email
303. [g_friends_daemon_ok](#) - Accept emails from any mailer deamon
304. [g_friends_debug1](#) - NEVER USE, only for NetWin testing
305. [g_friends_default_autoadd](#) - Default auto addition when sending (recommended)
306. [g_friends_default_mode](#) - Default friends mode (Recommended 'silent')
307. [g_friends_global_add](#) - Add to a global friends list if ip matches and sender doesn't match authenticated user
308. [g_friends_global_exclude](#) - Addresses not to auto add, e.g. *@paypal.com
309. [g_friends_ignore](#) - List of addresses considered friends for all users on the system
310. [g_friends_ignore_trusted](#) - If from trusted ip still apply friends
311. [g_friends_lang_auto](#) - Set users language settings automatically based on observed emails from friends
312. [g_friends_latest_headers](#) - Causes friends to re-read message headers, allowing rules based on headers added during delivery
313. [g_friends_long](#) - In friends web release addresses use a longer url
314. [g_friends_name](#) - What to call the friends system on pages and in email
315. [g_friends_old_status_email](#) - Use older status email & processing
316. [g_friends_only](#) - Enable friendly features - must be enabled by users too
317. [g_friends_pending_keep](#) - Number of days to store users pending messages before deleting them
318. [g_friends_pending_max](#) - Max items in pending before deleting them
319. [g_friends_pending_name](#) - The imap name of the friends_pending (and spam store) quarantine folder - should match surgeweb imap_spam_folder - default is 'Friends Pending'
320. [g_friends_pending_vanish](#) - Enable auto-vanish of pending messages on confirmation bounce
321. [g_friends_release_wash](#) - Clean any subject marking (ie stars) when releasing/allowing
322. [g_friends_rotate](#) - Rotate user level log file, default 30k
323. [g_friends_safer](#) - Make friends always avoid back scatter.
324. [g_friends_silent](#) - Disable friends responses to users
325. [g_friends_skip_ip](#) - List of ip addresses considered friends for all users on the system
326. [g_friends_spam_score](#) - Default level to quaranteen message in spam folder (Recommended 8 or 10)
327. [g_friends_spf_fail_bounce](#) - Bounce SPF failures, do not send friends confirmations (Not recommended)
328. [g_friends_url](#) - Specify default global url for friends release http://domain.name:port
329. [g_from_allow](#) - Other email addresses we allow, e.g. *@x.y.com,*@b.com,fred@bb.com
330. [g_from_allow_ip](#) - IP addresses to bypass local from check
331. [g_from_allow_to](#) - destination user to bypass local from check
332. [g_from_bl](#) - Domain Based Blacklist Zones, lookups FROM domain in dns
333. [g_from_bounce](#) - Reject if local from address is probably faked
334. [g_from_check](#) - Check from domains match valid local domains if user is authenticated, or g_from_allow
335. [g_from_domain](#) - Default domain for from envelope
336. [g_from_exact](#) - Check from matches authenticated user
337. [g_from_header](#) - From header used in delivery bounces
338. [g_from_must_exist](#) - Require local from addresses to exist or reject mail
339. [g_from_noforge](#) - If envelope or from is local domain then the other must be too
340. [g_from_noforge_some](#) - If from matches this then from/envelope must match
341. [g_from_noforgeme](#) - If to==from then from and env from must match
342. [g_from_nofriend](#) - If forge setting would bounce message then allow message but don't allow friend match
343. [g_from_relay](#) - If not authenticated and g_relay_allow_ip matched then block if not local domain or whitelisted
344. [g_from_relay_white](#) - White list of domains for g_from_relay setting
345. [g_from_rewrite](#) - Rewrite from envelope for outgoing email, e.g. *@this.domain -> %1@another.domain
346. [g_from_rewrite_header](#) - Rewrite the from header as well
347. [g_from_rewrite_sender](#) - Rewrite the sender header as well
348. [g_from_stamp](#) - Stamp if local from address is probably faked

- 349. [g_from_timeout](#) - Timeout when checking bad from addresses, default 60 seconds
- 350. [g_from_valid](#) - Require an @ and dotted domain in all return addresses
- 351. [g_gateway](#) - Gateway messages for that domain to the specified address
- 352. [g_gateway_allow](#) - Known hosts that act as incoming SMTP or surgewall servers for us
- 353. [g_gateway_always](#) - Always send to gateway even if local domain exists
- 354. [g_gateway_auth](#) - Send SMTP auth requests to another host
- 355. [g_gateway_data](#) - Gateway at the data stage
- 356. [g_gateway_from](#) - Pass 'from' header thru during gateway check
- 357. [g_gateway_helo](#) - Header that must exist in incoming bounces (g_send_helo) or bounces are dropped
- 358. [g_gateway_ifnot](#) - Send local deliveries to scanner (using gateway rule) before delivering locally, deliver locally if from ip matches
- 359. [g_gateway_ignorewild_ip](#) - Ignore * gateway rules if from ip matches (allows outbound email scanning using gateway * to external scanner)
- 360. [g_gateway_mx](#) - If specified IP address is found in mx record for destination then allow relay (not recommended)
- 361. [g_gateway_open](#) - Allows an open relay setting in g_gateway
- 362. [g_gateway_orcpt](#) - Writes an original receipt header when forwarding a message, this may disclose multiple recipients, cc/bcc etc use only for tracking faults
- 363. [g_group_field](#) - Auth field to add to group membership
- 364. [g_gzip_disable](#) - Disable gzip web compression
- 365. [g_hack_detect_disable](#) - Stop admin emails when users login with a weak password
- 366. [g_hack_msg](#) - Message to send to users with a weak password
- 367. [g_hack_touser](#) - Send warnings directly to users
- 368. [g_hacker_max](#) - Login guesses for one ip address before we lockout the ip address
- 369. [g_hacker_poison](#) - Poison accounts that instantly blacklist ip address e.g. root@*
- 370. [g_hacker_whitelist](#) - Ip addresses to avoid guessing issues
- 371. [g_header_out](#) - Header to add to outgoing posts
- 372. [g_header_strip](#) - Strip listed headers from incoming messages
- 373. [g_helo_optional](#) - Helo is optional for SMTP protocol (not recommended)
- 374. [g_help_local](#) - Make all help references to the local help files
- 375. [g_home](#) - Home path of server configs log etc
- 376. [g_honeypot_key](#) - Key for HTTP RBL service www.projecthoneypot.org - not recommended
- 377. [g_honeypot_rbl](#) - RBL name to lookup, typically dnsbl.httpbl.org
- 378. [g_host_redirect](#) - Redirection based on host for surgeweb's [https_required](#) redirection
- 379. [g_http_11](#) - Use http 1.1 requests to netwinsite (do not use)
- 380. [g_http_proxy](#) - Proxy web server for fetching files from netwinsite.com if direct access fails
- 381. [g_imap_acl](#) - Enable ACL (shared folders) in imap
- 382. [g_imap_auto_create](#) - Create folders matching this list in response to 'select' commands
- 383. [g_imap_blacklist](#) - Test if imap users are in rbl's and email admin
- 384. [g_imap_capa](#) - Where to get the CAPABILITY value from
- 385. [g_imap_capa_strip](#) - Capability values to hide
- 386. [g_imap_cram_enable](#) - Enable CRAM-MD5 authentication (requires [nwauth](#) 4.0h or greater)
- 387. [g_imap_debug](#) - For NetWin use only
- 388. [g_imap_delay](#) - Glob data into bigger packets, never use this
- 389. [g_imap_friends](#) - Make the friends_pending folder visible in imap
- 390. [g_imap_idle_free](#) - Experimental - releases threads in 'idle' state
- 391. [g_imap_idle_nsf](#) - The number of seconds before a complete directory rescan. To be used on NFS network drives
- 392. [g_imap_log_body](#) - Log imap fetch body commands to msg*.rec log files
- 393. [g_imap_log_copy](#) - Log imap copy commands to msg*.rec log files
- 394. [g_imap_log_flush](#) - Flush imap log on every write (for debugging)
- 395. [g_imap_log_header](#) - Log imap fetch header commands to msg*.rec log files (not usually needed)
- 396. [g_imap_log_protocol](#) - If true log IMAP protocol to log file
- 397. [g_imap_max_messages](#) - The number of messages in a single imap folder, default 200000
- 398. [g_imap_no_internal_date](#) - Disables internal date which helps stupid outlook client show correct dates
- 399. [g_imap_old](#) - Revert to old imap module
- 400. [g_imap_old_ip](#) - Revert to old imap module for some ip's
- 401. [g_imap_pop_burst](#) - Always burst using imap code
- 402. [g_imap_port](#) - IMAP port to listen on, default is 143
- 403. [g_imap_search_noattach](#) - Skip non text attachments when searching

- 404. [g_imap_secure_port](#) - IMAP SSL secure port to listen on, default is 993
- 405. [g_imap_size_fetch](#) - If true, will display message sizes on fetch command. (ie * 123 EXISTS)
- 406. [g_imap_spam_train](#) - Train if moving message to 'spam' folder, or from 'spam' folder to inbox
- 407. [g_imap_testing](#) - Test imap module instead of normal one (not functional)
- 408. [g_imap_throttle](#) - Limit for sustained imap commands per second before warning admin, default is 5
- 409. [g_imap_timeout](#) - Time, in minutes for imap timeout, RFC required default is 30
- 410. [g_imap_timezone](#) - Timezone to display - for testing purposes only NOT USED
- 411. [g_imap_uidl_nofix](#) - If true, disable auto repair of identical uidl entries
- 412. [g_imap_unsub_auto](#) - Unsubscribe if a folder doesn't exist
- 413. [g_imap_user_flags](#) - This setting may confuse some email clients (mac) use with caution
- 414. [g_inbox_max](#) - Max messages permitted in inbox e.g. 5000
- 415. [g_inbox_nolimit](#) - Users with no limit on inbox
- 416. [g_include](#) - Include another ini file global settings only
- 417. [g_iplimit](#) - Untrusted local ip addresses e.g. web servers, special sending limits applied.
- 418. [g_iplimit_islocal](#) - Add domains to list of domains considered local for limit counting
- 419. [g_iplimit_local](#) - Max sends from untrusted ip to local domains per 30 minutes.
- 420. [g_iplimit_remote](#) - Max sends from untrusted ip to remote domains per 30 minutes.
- 421. [g_iplimit_whitelist](#) - List of 'from' addresses that should bypass limits
- 422. [g_ipv6_enable](#) - Enable IPV6 networking only use if you have an IPV6 address for some reason
- 423. [g_keepalive](#) - Attempts to use keepalive for the web sessions (experimental & faulty currently)
- 424. [g_key_manual](#) - Try and activate automatically when the key expires
- 425. [g_key_nowarning](#) - Disable reminders to update your license
- 426. [g_known_skip](#) - Testing spf disable known bypass
- 427. [g_language_default](#) - Default language for user web interface
- 428. [g_last_login](#) - If true create last_login file each time user logs in via imap/pop. Do not use on MIRROR systems
- 429. [g_last_login_days](#) - If last login is more than this many days then reject email - do not use on mirrors
- 430. [g_late_forward](#) - Apply all users forwarding rules after friends, spam, and filtering
- 431. [g_ldap_forward](#) - Remote ldap server to forward requests to (only for testing do not use)
- 432. [g_ldap_outlook_browse_max](#) - Basic outlook ldap address browsing, max items (KEEP THIS SMALL eg <50): default=0 (disabled)
- 433. [g_ldap_port](#) - LDAP port, set to 389 to enable simple address book lookups only. (NOT YET FULLY FUNCTIONAL)
- 434. [g_legal_archive_add](#) - Users must belong to this group to get their email archived
- 435. [g_legal_archive_bucket](#) - bucket for for net service
- 436. [g_legal_archive_enable](#) - Enable legal archive
- 437. [g_legal_archive_encrypt_key](#) - Key for encrypting the data, you MUST never lose this
- 438. [g_legal_archive_hostid](#) - Unique integer for this host 1-9 use if sharing mail spool
- 439. [g_legal_archive_keep](#) - Days to keep legal archive, units=days unless you specify years or months, default 5 years
- 440. [g_legal_archive_local](#) - Store files locally only
- 441. [g_legal_archive_path](#) - Local path for archive indexes
- 442. [g_legal_archive_show](#) - Users must belong to 'archive_show' group to see their own archive
- 443. [g_if_fix_off](#) - If input contains naked 'lf' characters then reject with error instead of stripping as usual
- 444. [g_local_skipgateway](#) - If true skip gateway rule for local messages (bounces etc)
- 445. [g_log_bounce_disable](#) - Stop bounce reject entries filling up log (typically from spam bounces)
- 446. [g_log_date](#) - Log full date in log files
- 447. [g_log_date_msg](#) - Log full date in msg log files (g_log_date required too)
- 448. [g_log_disable](#) - Disable most logging - not recommended
- 449. [g_log_dns](#) - Log dns responses in gory detail
- 450. [g_log_dropped_disable](#) - Don't log if no 'data' command sent
- 451. [g_log_flush](#) - Flush log file after every write
- 452. [g_log_fwd](#) - This setting is obsolete and has no effect
- 453. [g_log_level](#) - Level of logging, info, debug, error
- 454. [g_log_norcpt](#) - Disable Log individual recipients in msg.rec files
- 455. [g_log_password](#) - Log password failures to login_failed.log
- 456. [g_log_path](#) - Directory for log files, defaults to G_HOME
- 457. [g_log_pid](#) - Log PID in log lines
- 458. [g_log_quota](#) - Log quota for specified user
- 459. [g_log_reject_disable](#) - If true then rejects are not recorded in .rec files

- 460. [g_log_size](#) - Size of each mail*.log file (e.g. 5mb)
- 461. [g_log_slow](#) - Do slower logging system
- 462. [g_log_start_norotate](#) - Don't rotate log on startup
- 463. [g_log_syslog](#) - Send 'msg.rec' entries to syslog
- 464. [g_log_syslog_debug](#) - Send 'mail.log' entries to syslog as 'mail.debug' data
- 465. [g_log_syslog_host](#) - Specify host to send syslog entries to (windows only)
- 466. [g_log_syslog_only](#) - Disable writing to msg.rec
- 467. [g_log_tcp_read](#) - Log actual tcp read data - for matching ip addresses - avoid
- 468. [g_log_tcp_write](#) - Log actual tcp write data - for matching ip addresses - avoid
- 469. [g_log_thid](#) - Log thread id in .rec files
- 470. [g_log_user](#) - Log pop/imap/smtp protocol for specified user
- 471. [g_lookup_names](#) - Lookup ip names of connecting users (can be slow)
- 472. [g_lookup_reject_fails](#) - If lookup cannot get a name, reject user (not generally recommended)
- 473. [g_lowdisk_warning](#) - Disksize below which to send a warning to the system manager
- 474. [g_mailbox_inbox](#) - Path for inboxes (experimental, do not use!)
- 475. [g_mailbox_path](#) - Default directory to store mail
- 476. [g_maildir_max](#) - Max messages in a folder, do not adjust
- 477. [g_maildir_netwin](#) - Use NETWIN proprietry storage format - Not Recommended
- 478. [g_maildir_report](#) - Email manager on ndb errors
- 479. [g_maildir_standard](#) - Use more standard maildir layout (NOT SUPPORTED)
- 480. [g_mailstatus_message](#) - Error message to give when mailstatus is set to specified state
- 481. [g_manager](#) - Email address of manager
- 482. [g_manager_port](#) - HTTP Manager port to listen on, default is 7026
- 483. [g_manager_secure_port](#) - HTTPS secure Manager port, default is https 7025
- 484. [g_manager_smtp](#) - SMTP server for error reporting
- 485. [g_manager_username](#) - Global domain managers username (for web based domain administration)
- 486. [g_max_bad_ip](#) - Max bad recipients per ip address before blocking that ip
- 487. [g_max_bad_ip_skip](#) - Skip g_max_bad_ip tests
- 488. [g_max_bad_ip_time](#) - Seconds to block guessing hackers
- 489. [g_max_bad_nlookup](#) - Max bad recipients in a row, if exceeded skip user lookup
- 490. [g_max_bad_to](#) - Max bad recipients in a row
- 491. [g_mdir_hash](#) - Hashing mode for surgemail (not supported, use at your own risk)
- 492. [g_mdir_prefix](#) - Prefix for maildir folders DO NOT USE THIS SETTING, NOT SUPPORTED!!!
- 493. [g_mfilter_addonly](#) - If true, then only allow 'adding' headers, not changing them
- 494. [g_mfilter_bounces](#) - Run mfilter on bounce messages and responders etc
- 495. [g_mfilter_file](#) - Mfilter rule file. For spam rule processing (mfilter.rul)
- 496. [g_mfilter_localonly](#) - If true then only run mfilter on local deliveries
- 497. [g_mfilter_maxlen](#) - Size to truncate messages to before processing with filter
- 498. [g_mfilter_noiseey](#) - Do log anything in mfilter
- 499. [g_mfilter_skip_from](#) - From addresses (envelope) to skip mfilter processing for
- 500. [g_mfilter_skip_ip](#) - IP address(es) to skip mfilter processing for
- 501. [g_mfilter_skip_to](#) - To addresses to skip mfilter processing for
- 502. [g_mfilter_trace](#) - Log trace lines in mfilter
- 503. [g_migrate_skip](#) - Skip imap folders matching this, use for shared folders
- 504. [g_mirror_config](#) - Mirror surgemail.ini to/from mirror_host
- 505. [g_mirror_config_except](#) - Settings to ignore when accepting the incoming config
- 506. [g_mirror_debug](#) - Log more info to mirror log.
- 507. [g_mirror_email](#) - Email manager list of fixes sent
- 508. [g_mirror_host](#) - Mirror other host name
- 509. [g_mirror_live](#) - Mirror: Send incoming messages immediately
- 510. [g_mirror_live_max](#) - Limit size of mirror_live default 60k
- 511. [g_mirror_mode](#) - Mirroring mode (one system must be MASTER and the other SLAVE)
- 512. [g_mirror_nossl](#) - Disable SSL for mirror protocol connection - recommended
- 513. [g_mirror_nwauth](#) - Mirror send nwauth database to other server, ONLY set on master
- 514. [g_mirror_nwauth_always](#) - Mirror nwauth database files
- 515. [g_mirror_prune_age](#) - Mirror minimum age for items to be pruned during sync_prune, default 14 days
- 516. [g_mirror_repair](#) - Run resync_prune each night, only set on master
- 517. [g_mirror_secret](#) - Mirror shared secret
- 518. [g_mirror_threads](#) - Max threads we can use during resync_fast, default 6
- 519. [g_mirror_trash](#) - Normally on a resync the trash folder is ignored.

- 520. [g_monitor_disable](#) - Disable monitor process completely (requires restart)
- 521. [g_monitor_port](#) - HTTP port for Surgemail Monitor to listen on, default is 7027
- 522. [g_msg_hops_max](#) - Maximum received lines or message is bounced, default 30
- 523. [g_msg_log_extra](#) - Extra user activity logging
- 524. [g_msg_log_from](#) - Log From in msg*.rec
- 525. [g_msg_max](#) - Max size of a single message (if over refuse with 552 error)
- 526. [g_msg_max_drop](#) - Drop link if size exceeded instead of waiting for the message to all arrive
- 527. [g_msg_max_total](#) - Max size of a message * recipients
- 528. [g_msg_track](#) - Message tracking - for debugging
- 529. [g_mutex_fast](#) - Use fast mutex handling DEBUGGING option only
- 530. [g_mutex_timeout](#) - Default mutex timeout period in seconds default is 600
- 531. [g_mutex_timing](#) - Name of mutex to collect extra timing information for
- 532. [g_mx_tryall](#) - Try all mx hosts even if lower than own mx priority
- 533. [g_myrbldisable](#) - Disable internal rbl database
- 534. [g_myrbldisable_rbl](#) - Disable netwin rbl database
- 535. [g_myrbldisable_fake](#) - Fake myrbldisable response for testing
- 536. [g_myrbldisable_share](#) - Use and Share RBL reputation data with central NetWin server (Recommended)
- 537. [g_myrbldisable_store](#) - Size of internal myrbldisable database
- 538. [g_myrbldisable_to](#) - Debug setting for rbl sharing do not use
- 539. [g_myurl_disable](#) - Disable internal url database
- 540. [g_naked_msg](#) - Error message if body contains naked If characters
- 541. [g_newui_advanced](#) - Always run new admin ui in advanced mode
- 542. [g_newui_disable](#) - Disable new admin ui (do not use)
- 543. [g_no_bull](#) - Special accounts that should not get bulletins
- 544. [g_notag_notascii](#) - Don't add x-notascii: charset to any non ascii message
- 545. [g_notag_url_forgery](#) - Don't add x-UrlForgery when a ref urls seem to not match
- 546. [g_nwv_test](#) - Test NetWin setting, best not played with)
- 547. [g_old_imap_headbody](#) - Get head and body seperately
- 548. [g_old_pophost_debug](#) - Log extra info when doing old pophost logins
- 549. [g_old_user_check](#) - Disable the account status enabled check on rcpt lines
- 550. [g_old_webmail_links](#) - Show webmail links in user cgi instead of surgeweb
- 551. [g_orbs_cache_life](#) - Time to keep RBL cached entries in seconds, default is 7200 seconds
- 552. [g_orbs_check_all](#) - Keep doing lookups even if found in a RBL, this is slower of course!
- 553. [g_orbs_exception](#) - Realtime Blackhole List, exception list of IP addresses
- 554. [g_orbs_fake](#) - Ip address to pretend we find in rbl database for testing
- 555. [g_orbs_force](#) - Force RBL check even if g_allow_ip matches this ip number
- 556. [g_orbs_late](#) - Do late disconnect so user has time to send SMTP authentication (Also applies G_SPF_SKIP_TO)
- 557. [g_orbs_list](#) - Realtime Blackhole Lists (RBL's), action=deny,accept,stamp
- 558. [g_orbs_rec](#) - Log to record file if RBL deny action occurs (can fill logs up)
- 559. [g_orbs_report](#) - List of IP's to check in RBL(s)
- 560. [g_orbs_service](#) - Service Name - Obsolete - use g_orbs_list to define services
- 561. [g_orbs_submit](#) - Do orbs check when 'data' command is sent or first valid recipient
- 562. [g_orbs_system](#) - If true use system dns lookups instead of surgemails for orbs (not recommended)
- 563. [g_orbs_testing](#) - If true, RBL lookups are recorded but not blocked
- 564. [g_orbs_timeout](#) - Seconds to wait for RBL lookups, default is 10 seconds
- 565. [g_outgoing_n](#) - Send manager email if more than this many spam from one user per day
- 566. [g_outgoing_white](#) - Whitelist for outgoing spam detector
- 567. [g_perflog_disable](#) - Completely disable 'perflog' historical performance logging
- 568. [g_perflog_flush_interval](#) - Interval in seconds to flush the performance log files to disk (default 1hr = 3600)
- 569. [g_perflog_logall](#) - Log all trend graph counters including undisplayed graphs (recommended)
- 570. [g_perflog_lowres](#) - Do low resolution perflog sampling (hiding hour scale)
- 571. [g_perflog_surgeonly](#) - Only log surgemail counters
- 572. [g_pipelining](#) - Show pipelining in ehlo response - not recommended - has no behavior affect
- 573. [g_pop_add_size](#) - Improves pop performance on nfs slightly
- 574. [g_pop_blocksize](#) - Size of packets to read pop messages (best left alone)
- 575. [g_pop_delay](#) - If true packets are sent in bunches, this slows down some mail clients
- 576. [g_pop_flush_lines](#) - Fluch to tcp every line of message sent (slow)
- 577. [g_pop_lock](#) - Lock pop spool
- 578. [g_pop_max](#) - Max threads for POP or IMAP connections

- 579. [g_pop_min_late](#) - Give min time error on first command after login
- 580. [g_pop_min_msg](#) - Additional warning to give user when they login too soon
- 581. [g_pop_min_skip](#) - Skip ip addresses matching this list.
- 582. [g_pop_min_time](#) - Min time in seconds between consecutive POP logins, NEVER USE
- 583. [g_pop_nolock](#) - Allows concurrent pop logins, recommended
- 584. [g_pop_port](#) - POP3 port to listen on, default is 110
- 585. [g_pop_secure_port](#) - POP3 SSL secure port to listen on, default is 995
- 586. [g_pop_warning](#) - Send manager warning if this many sessions (pop or imap) reached (max 1 per hour)
- 587. [g_popfetch](#) - Fetch incoming mail from another pop server
- 588. [g_popfetch_interval](#) - Interval between popfetch attempts in seconds
- 589. [g_popfetch_kick](#) - If true then popfetch will try and open the link for 10 seconds, then retry, this should bring up ISDN lines.
- 590. [g_popfetch_nodup](#) - Drop duplicate messages
- 591. [g_ppd_port](#) - PopPassD port for setting passwords, default is 106
- 592. [g_private](#) - Enable a private customer specific feature
- 593. [g_proxy](#) - Proxy mode, best avoided for most situations
- 594. [g_proxy_default](#) - Proxy mode default forward to host
- 595. [g_proxy_to_gateways](#) - Proxy pop/imap connections to matching gateway settings
- 596. [g_proxy_usercgi](#) - Proxy user.cgi requests to tohost (web_ref_text.txt & g_web_ref_path_extension must match on all servers)
- 597. [g_proxy_webmail](#) - Redirect webmail logins to external host name
- 598. [g_pstat_disable](#) - Disable pstat per user accounting (for debugging)
- 599. [g_queue_limit](#) - If on disk queue exceeds this block incoming mail
- 600. [g_queue_max](#) - Size of internal que file cache, range 500-3000
- 601. [g_queue_warning](#) - If on disk queue exceeds this send manager a warning
- 602. [g_quota](#) - Disk quota for users in specified g_access_group
- 603. [g_quota_disable](#) - Disable quota system
- 604. [g_quota_friends](#) - Count friends pending messages as part of quota
- 605. [g_quota_noemail](#) - Disables all quota messages to the user
- 606. [g_quota_rcpt_disable](#) - Disables quota check at rcpt stage
- 607. [g_quota_report](#) - Send quota warnings to the manager
- 608. [g_quota_skip](#) - Skip quota for matching ip addresses
- 609. [g_quota_try_later](#) - Give 450 response if user is over quota so message will be resent
- 610. [g_quota_warning_disable](#) - Disables the 80% quota warning message
- 611. [g_rcpt_bang](#) - Allow bang character in addresses
- 612. [g_rcpt_colon](#) - Allow colon character in addresses
- 613. [g_rcpt_max](#) - Max recipients per message, default 1000, can only be lower than 3000.
- 614. [g_rcpt_max_in](#) - Limit for recipients of untrusted channels, default g_rcpt_max
- 615. [g_rcpt_msg](#) - Response given for invalid recipient errors, message is prefixed by email address.
- 616. [g_rcpt_nodup](#) - Ignore duplicate recipients to the same user
- 617. [g_rcpt_quote](#) - Allow quote character(s) in addresses
- 618. [g_rcpt_trace](#) - Add X-Rcpt-Trace headers
- 619. [g_rdns_timeout](#) - Timeout for reverse DNS lookups default is 30 seconds
- 620. [g_received_name](#) - Name shown in received headers
- 621. [g_received_names](#) - List of valid received names for incoming email
- 622. [g_received_skip](#) - Skip local received header for trusted users (authenticated or g_relay...)
- 623. [g_received_skip_all](#) - Skip local received header for messages that have non local recipients
- 624. [g_received_skip_spf](#) - Skip spf received header for messages that have non local recipients
- 625. [g_recent_bypass](#) - Bypass recent failure checking
- 626. [g_record_days](#) - Days to keep record of incoming messages, default 90
- 627. [g_record_hash](#) - Hash storage of daily .rec files
- 628. [g_record_path](#) - Directory for daily .rec files defaults to G_HOME
- 629. [g_redirect](#) - Redirect messages from 'was' to the new 'to' address
- 630. [g_redirect_cc](#) - Send carbon copy to another address
- 631. [g_redirect_cc_attach](#) - Redirect message as attachment if rule applies
- 632. [g_redirect_from](#) - Redirect if from envelope matches
- 633. [g_redirect_from_cc](#) - Send carbon copy if from envelope matches
- 634. [g_redirect_hide](#) - Hide the redirection in the output
- 635. [g_redirect_iflocal](#) - If local domain, then apply redirect
- 636. [g_redirect_ignore_errors](#) - Accept email even if redirected addresses fail

- 637. [g_redirect_noautocreate_rules](#) - Don't create redirection rules for domains automatically
- 638. [g_redirect_ses](#) - If message is not local then apply redirect
- 639. [g_relay_allow_from](#) - Allow relaying from users if the from envelope and from header match this NEVER USE, SPAMMERS ABUSE THIS
- 640. [g_relay_allow_ip](#) - Allow relaying from users at this ip address
- 641. [g_relay_dom_and_ip](#) - Allow relaying if from envelope and ip address both match
- 642. [g_relay_ifnot](#) - Accept locally only if not from this ip
- 643. [g_relay_message](#) - Message to give to users who try to relay through your system
- 644. [g_relay_nolocal](#) - Do not automatically relay for 127.0.0.1
- 645. [g_relay_process](#) - Relay process, e.g. testip.exe \$WHOIP, return 1 to allow relaying, 0=deny
- 646. [g_relay_to](#) - Relay to this domain from anyone
- 647. [g_relay_to_user](#) - Relay to specific user from anyone
- 648. [g_relay_window](#) - Minutes to allow relay after pop/imap login NOT RECOMMENDED
- 649. [g_relay_window_from](#) - Requires pop authed user is in from header of sent message
- 650. [g_rename_files](#) - Wild card list of files to rename, e.g. *.exe,*.cmd (see help for defaults)
- 651. [g_report_host](#) - Report facts to a central host
- 652. [g_report_notspam](#) - Send not spam samples to netwinsite.com automatically
- 653. [g_report_spam](#) - Send spam samples to netwinsite.com automatically
- 654. [g_responder_delay](#) - Delay between responses to the same address.
- 655. [g_responder_from](#) - Send 'from' destination user
- 656. [g_responder_safer](#) - Only respond if the sender can be verified in some way (spf/domainkeys)
- 657. [g_responder_score](#) - Respond anyway if spam score is below this (default 10)
- 658. [g_responder_sender](#) - Responder whitelist for email from address
- 659. [g_responder_skip](#) - Skip responder if from matches
- 660. [g_responder_source](#) - Responder whitelist for from ip name or number
- 661. [g_responder_to](#) - Responder whitelist for destination user
- 662. [g_responder_utf8](#) - Send response in utf8 format
- 663. [g_restart](#) - Restart server if it dies
- 664. [g_retry_bounces](#) - Max hours to keep trying to deliver a bounce, default is 48hrs
- 665. [g_retry_dns](#) - Hours to keep trying if dns response suggested invalid domain name, default 0
- 666. [g_retry_from](#) - Time to keep messages from these domains
- 667. [g_retry_limit](#) - Max hours to keep trying to deliver a message, default is 48hrs
- 668. [g_retry_minutes](#) - Time between attempting resends, defaults to 60 minutes
- 669. [g_retry_rule](#) - Time to keep messages to these domains
- 670. [g_retry_unwarn](#) - Send user sent on confirmation if warning sent
- 671. [g_retry_warn](#) - Send user a warning if first send fails
- 672. [g_retry_warn_n](#) - Send user a warning if nth send fails
- 673. [g_route](#) - Route messages matching particular wildcard spec to specified server
- 674. [g_route_by_tohost](#) - Route messages using server specified in 'tohost' in authent database
- 675. [g_route_except](#) - IP exception to g_route / g_route_by_tohost
- 676. [g_route_local](#) - Route messages for local domains if the rule applies
- 677. [g_sabre_version](#) - SabreDAV version (DO NOT CHANGE, for debugging only)
- 678. [g_safe_imap](#) - Force users to prove they are real if logging in from pop/imap NEVER NEVER USE
- 679. [g_safe_smtp](#) - Force users to prove they are real if logging in from unknown sources via smtp
- 680. [g_safe_smtp_email](#) - Email manager as remote ip addresses are added
- 681. [g_safe_white](#) - White list for g_safe* settings
- 682. [g_sample_get](#) - Sample account to check if deliveries work
- 683. [g_sample_show](#) - Headers to show from sample messages
- 684. [g_scan_action](#) - Converts return value from g_scan_cmd, action=drop,accept,bounce
- 685. [g_scan_cmd](#) - Run command on message, and return integer, see g_scan_action
- 686. [g_scan_cmd_skip](#) - Skip for matching ip addresses
- 687. [g_scan_cmd_testing](#) - Don't reject, (for testing)
- 688. [g_sched_utoken_timeout](#) - Timeout for sched utokens in minutes
- 689. [g_send_backoff](#) - Seconds to leave slow responding host alone (default 900)
- 690. [g_send_body_end_retry](#) - Try again if connection fails after entire body sent
- 691. [g_send_body_noretry](#) - If a send fails during the body send give up at once.
- 692. [g_send_body_once](#) - Don't try 3 times if failure occurs sending body
- 693. [g_send_conspeed](#) - Outgoing connections per second per destination, default is 4
- 694. [g_send_delay](#) - Wait this many seconds after sending each item.
- 695. [g_send_first_retry](#) - Minutes for first retry, default is 16 minutes, do not adjust!

- 696. [g_send_helo](#) - Fully qualified domain to use for all outgoing SMTP helo commands and MessageIDs
- 697. [g_send_helo_from](#) - Use matching domain name if we have one if user is authenticated/trusted AVOID THIS!
- 698. [g_send_helo_in](#) - Lookup dns name of incoming ip connection on local interface
- 699. [g_send_lines](#) - Send messages in single line packets, slow!
- 700. [g_send_lowpriority](#) - Ip address of bulk sending servers
- 701. [g_send_max](#) - How many concurrent sending sessions in total
- 702. [g_send_max_perchan](#) - Msgs to send on one open channel
- 703. [g_send_max_perdom](#) - How many concurrent sessions allowed to another domain, default is 2
- 704. [g_send_max_rcpt](#) - How many rcpt's to send per message when sending
- 705. [g_send_no_domain](#) - Message to show when domain points to us but can't find user or domain
- 706. [g_send_nolimit](#) - Don't apply g_max_perdom limit when sending to this domain
- 707. [g_send_nopoll](#) - Use sleep loop instead of poll (debugging only)
- 708. [g_send_noskipslow](#) - Don't remember hosts that are slow to open and avoid them
- 709. [g_send_onpopfetch](#) - Only send outgoing while doing a popfetch (For dialup use)
- 710. [g_send_open_timeout](#) - Timeout, in seconds when opening a link
- 711. [g_send_retry_552](#) - Retry on 552 responses (typically quota exceeded)
- 712. [g_send_rewrite](#) - Rewrite envelope recipient at send stage, does not change destination server
- 713. [g_send_speed](#) - Bytes per second to limit each outgoing channel to, default no limit, eg 10k
- 714. [g_send_store_disable](#) - Disable sendstore smtp extension
- 715. [g_send_timeout](#) - Timeout, in seconds when sending, default is 540 (9 minutes)
- 716. [g_send_tolimit](#) - Limit speed to send to one or more domains.
- 717. [g_sent_store](#) - Store all sent messages in a folder if smtp authenticated
- 718. [g_server_name](#) - SERVER_NAME to set for list of wildcard urls
- 719. [g_server_stamp](#) - Replaces SurgeMail and version string in received headers
- 720. [g_setpassword_firstlogin](#) - Accept any password on first POP login and set in database (EMERGENCY USE ONLY, requires nwauth -reasonfail parameter)
- 721. [g_sf_disable](#) - Smart Filter Disable
- 722. [g_sf_generate](#) - Build local smart filter
- 723. [g_sf_ignore_users](#) - Ignore user submissions just use automatic samples
- 724. [g_sf_nnet](#) - Use Neural Network (Experimental, ONLY FOR TESTING)
- 725. [g_sf_test2](#) - Testing
- 726. [g_share_home](#) - Allow sharing of home directory
- 727. [g_share_mail](#) - Set true if mail area is shared (by nfs or other mechanism)
- 728. [g_share_quota](#) - Do quota on disk (e.g. when using nfs shared spool)
- 729. [g_shutdown_slow](#) - Add 20 second delay to shutdown for debugging
- 730. [g_slow_welcome](#) - Add 30 second delay to welcome message for debugging
- 731. [g_smite_all](#) - Add spamdetect and smitematch headers to all messages going past
- 732. [g_smite_gateway](#) - Add spamdetect and smitematch headers to gatewayed/redirected messages
- 733. [g_smite_level](#) - If smitematch score is above this drop message (just throw it away) e.g. 1
- 734. [g_smite_skip](#) - Whitelist/Skip spam scanner (and spf) if from matches this wild card (Whitelist)
- 735. [g_smite_skip_auth](#) - Skip spam scanner if user logged in
- 736. [g_smite_skip_ip](#) - Skip spam scanner if senders ip matches
- 737. [g_smite_skip_only](#) - Skip spam scanner if to matches this wild card and no other recipients that 'don't' match...
- 738. [g_smite_skip_relay](#) - Skip spam scanner if ip can relay
- 739. [g_smite_skip_to](#) - Skip spam scanner if to matches this wild card
- 740. [g_smite_tag](#) - Tag message with smitematch header if message is in spam database when read
- 741. [g_sms_forward](#) - Specifies IP's which are allowed to forward to SMS gateways
- 742. [g_sms_gateway](#) - Address and port of your sms gateway
- 743. [g_sms_gateway_force](#) - Force sms notifications to go to g_sms_gateway
- 744. [g_sms_gateway_msgbytes](#) - Maximum amount of message to send to g_sms_gateway (bytes)
- 745. [g_sms_gateway_subjbytes](#) - Maximum length of subject in sms message
- 746. [g_smtp_auth_debug](#) - Auth Debug (do not use)
- 747. [g_smtp_auth_ip](#) - Ip Addresses to accept smtp authentication from
- 748. [g_smtp_auth_off](#) - Disable SMTP AUTH from unknown ip addresses
- 749. [g_smtp_big](#) - Slow down incoming SMTP reads to get bigger packets (experimental)
- 750. [g_smtp_bounce_nslow](#) - Number of handles to use for doing slow rejections of smtp connections
- 751. [g_smtp_cmd_timeout](#) - Seconds to wait after getting a message for next command (sendmail bug)
- 752. [g_smtp_cram_enable](#) - Enable CRAM-MD5 authentication (requires nwauth 4.0h or greater) - Not Recommended

- 753. [g_smtp_data_timeout](#) - Seconds for timeout for data input, default 540 (9 minutes)
- 754. [g_smtp_delay_stamp](#) - Stamp header if sender sends data before seeing welcome response (usually spam)
- 755. [g_smtp_etrn_auth](#) - Only do etrn processing if user is authenticated
- 756. [g_smtp_fast_bounce](#) - Reject bad connections immediately
- 757. [g_smtp_fix_nohead](#) - Accept messages with no headers and try and cope
- 758. [g_smtp_help_disable](#) - Disable help in SMTP (minor security issue)
- 759. [g_smtp_log_protocol](#) - If true log SMTP protocol to log file
- 760. [g_smtp_log_size](#) - Size of smtp.log file
- 761. [g_smtp_max](#) - Max concurrent incoming SMTP connections
- 762. [g_smtp_max_nolimit](#) - Ip addresses that don't have max smtp limit applied
- 763. [g_smtp_max_reason](#) - Reason to give to user if g_smtp_max is exceeded
- 764. [g_smtp_maxbad](#) - Max bad command per session before dropping smtp link, default no limit
- 765. [g_smtp_no_brackets](#) - Allow from/rcpt without angle brackets
- 766. [g_smtp_noauth](#) - Accept incoming SMTP from these IPs (other IPs allowed if authenticated), default is *
- 767. [g_smtp_noauth_msg](#) - Message given when sender is told to use authentication because of g_smtp_noauth
- 768. [g_smtp_noauthm](#) - Accept incoming SMTP from these IPs (multi line version of g_smtp_noauth), default is *
- 769. [g_smtp_plain_hide](#) - Hide 'plain' from the ehlo response
- 770. [g_smtp_port](#) - SMTP port to listen on, default is 25
- 771. [g_smtp_portauth](#) - SMTP ports which require smtp authentication, typically 587
- 772. [g_smtp_portforce](#) - Block logins for ports not listed in g_smtp_portauth
- 773. [g_smtp_secure_port](#) - SMTP SSL secure port to listen on, default is 465
- 774. [g_smtp_thread](#) - Use separate thread for incoming SMTP connections
- 775. [g_smtp_vrfy_msg](#) - Change Response to VRFY, e.g. 252 Not telling
- 776. [g_smtp_warning](#) - Send manager warning if this many sessions reached (max 1 per hour)
- 777. [g_smtp_welcome_delay](#) - Seconds to delay welcome message, drop if we get data before we send welcome, recommend 1-3 seconds
- 778. [g_spam_alias_any](#) - User alias string e.g. "++" if defined then strip suffix from emails - not advised!
- 779. [g_spam_allow](#) - IP Wild card exceptions to spam limits
- 780. [g_spam_allow_disable](#) - Disable allow bounce messages
- 781. [g_spam_allow_known](#) - Unblock IP address if we have received messages from it for 3 days (so it's not a transient spammer)
- 782. [g_spam_allow_msg](#) - Template for unblock messages, use ||reason|| and ||allow|| and maybe a url
- 783. [g_spam_allow_rbl](#) - Give unblock message to RBL bounces too
- 784. [g_spam_allow_recent](#) - Skip spam rules if recent pop ip number
- 785. [g_spam_aspam](#) - Scale for aspam, default is 1.0, Valid range is zero to two
- 786. [g_spam_autotrain](#) - Auto train spam filter good messages based on first 1000 outgoing emails
- 787. [g_spam_black_auto](#) - Auto blacklist for user when isspace pressed
- 788. [g_spam_block](#) - Block spam (as decided by spf etc), if not set then user or domain can set
- 789. [g_spam_block_gateway](#) - Block spam gatewayed messages too
- 790. [g_spam_block_msg](#) - Template for spf blocked message if allow is disabled
- 791. [g_spam_body](#) - If spamdetect score is above this, add spamdetect header at top of message body NOT RECOMMENDED e.g. 7
- 792. [g_spam_body_more](#) - Add more info to spam body (ip address, ptr address, reply to and bounce address)
- 793. [g_spam_body_url](#) - Text part of info to add to body, usually a url to your site
- 794. [g_spam_bounce](#) - If spamdetect score (number of '*'s) is above this, bounce message if local delivery. NEVER USE THIS
- 795. [g_spam_bounce_all](#) - If spamdetect score is above this, bounce message, applies to all messages regardless of user settings. e.g. 7 NEVER USE THIS
- 796. [g_spam_bounce_text](#) - Error to return to user when message is bounced due to g_spam_bounce setting
- 797. [g_spam_bounce_trusted](#) - If spamdetect score is above this, bounce message if trusted (spam_allow or authenticated)
- 798. [g_spam_catcher](#) - Addresses on web pages that shouldn't get any email (robot bait)
- 799. [g_spam_char](#) - Character to use instead of '*' for smitespam headers (best left alone if possible)
- 800. [g_spam_check_auth](#) - Don't skip spam rules for authenticated users
- 801. [g_spam_cmd](#) - Command line spam checker, use \$FILE\$ in cmd parameters
- 802. [g_spam_cmd_if](#) - If internal spam rating is below this number, then run external filter
- 803. [g_spam_cmd_reject](#) - If external filter returns number larger than this reject
- 804. [g_spam_cmd_skip](#) - If internal spam rating is below this number, then skip external filter
- 805. [g_spam_content_disable](#) - Disable aspam_content.txt rules
- 806. [g_spam_flag](#) - Add X-SPAM-FLAG: Yes header if smite score is above this level

- 807. [g_spam_folders](#) - Train on any message dropped into the relevant folders
- 808. [g_spam_folders_show](#) - List the special folders for all users
- 809. [g_spam_from_blacklist](#) - Fetch list of bad domains to reject email from - not recommended
- 810. [g_spam_from_max](#) - Max outgoing messages per ipaddress/return path pair, 30 minutes, e.g. 5000
- 811. [g_spam_grey](#) - OBSOLETE DO NOT USE, Enable old greylisting for spf mechanism
- 812. [g_spam_grey_bounce](#) - Bounce if message was allowed due to grey listing, and spam score is above this, default 8 (was 4)
- 813. [g_spam_grey_classc](#) - Apply grey listing to x.x.x.*
- 814. [g_spam_grey_dflt](#) - Enable greylisting for spf default accept events (not recommended)
- 815. [g_spam_grey_dflt_bad](#) - Enable greylisting instead of allow in some cases (recommended for block or strict)
- 816. [g_spam_grey_nofive](#) - Skip 5-6 minute black window for these domains
- 817. [g_spam_grey_nohard](#) - Avoid hard spf bounces always try and do a grey list instead
- 818. [g_spam_grey_nseen](#) - Number of messages from an unknown host, default is 6
- 819. [g_spam_grey_size](#) - Size of grey listing table, default is 3000
- 820. [g_spam_grey_verify](#) - Skip grey listing if host was not listening
- 821. [g_spam_grey_window](#) - Window to block bad messages, typically 60 seconds
- 822. [g_spam_header_trust_ip](#) - List of IP addresses from which to trust/accept existing X-SpamDetect headers in emails
- 823. [g_spam_hold_hide](#) - Hide spam hold settings for end users and other held2pend user.cgi tweaks
- 824. [g_spam_hold_keep](#) - Number of days to store users spam hold messages before deleting them
- 825. [g_spam_info](#) - Info line and url to explain aspm system
- 826. [g_spam_info_hide](#) - Removes the x-spamdetect-info header line
- 827. [g_spam_internal](#) - Enable new 'internal' spam processing
- 828. [g_spam_isspam_ignore](#) - Don't block messages from ip addresses recorded as a spam source
- 829. [g_spam_isspam_kind](#) - Allow isspam from recent pop, gateway to etc
- 830. [g_spam_nobounce](#) - Remove old user held/vanish but after 5.2 will allow bounce
- 831. [g_spam_nolang](#) - Don't add header with a guess at body language
- 832. [g_spam_notrain](#) - Disable isspam and notspam addresses for user training
- 833. [g_spam_notspam](#) - Address that non authenticated users can send non spam to.
- 834. [g_spam_noupdate](#) - Disable fetch of aspm filter rules etc from netwinsite
- 835. [g_spam_phishing](#) - Download list of known phishing addresses and block outgoing email to them
- 836. [g_spam_phrase](#) - Enable auto spam phrase filter
- 837. [g_spam_poly](#) - Scale for poly word matching, default is 0.1, Valid range is zero to two, Use 1.0 to enable, EXPERIMENTAL
- 838. [g_spam_poly_disable](#) - Disable poly code.
- 839. [g_spam_private](#) - Enable users to define 'private' extensions user--STUFF@domain
- 840. [g_spam_probe](#) - Probe suspect urls to find spammers
- 841. [g_spam_probe_friends](#) - Probe even if email is from a friend
- 842. [g_spam_probe_more](#) - Probe even if email is from a known ip address
- 843. [g_spam_probe_unknown](#) - Probe any unknown url (dangerous)
- 844. [g_spam_probe_whois](#) - Do whois lookups on web pages found in probe
- 845. [g_spam_share](#) - Use and share some spam/aspm information with central server (netwin) experimental
- 846. [g_spam_status_hour](#) - Process all spam status messages at this time (disk io intensive)
- 847. [g_spam_status_monthly](#) - Send monthly spam status even if no messages pending
- 848. [g_spam_subject](#) - If score is above this, add spam rating to subject (Spam: ****) e.g. 8
- 849. [g_spam_subject_dom](#) - Destination domains to tag subject for
- 850. [g_spam_subject_gateway](#) - If true then spam_subject setting applies to gatewayed messages too
- 851. [g_spam_subject_word](#) - The word that gets added to subject, default is 'Spam', UCE is another good one
- 852. [g_spam_url](#) - Scale for url word matching, default is 1.0, Valid range is zero to two
- 853. [g_spam_user_max](#) - Max messages an authenticated user can send per 30 minutes, e.g. 5000
- 854. [g_spam_user_skip](#) - Users to skip g_spam_user_max limit for
- 855. [g_spam_userconfig](#) - Allow users to specify specific spam features
- 856. [g_spam_vanish](#) - If spamdetect score (number of '*'s) is above this, vanish message if local delivery. NEVER USE THIS
- 857. [g_spam_vanish_all](#) - If spamdetect score is above this, vanish message, applies to all messages regardless of user settings. NEVER USE THIS
- 858. [g_spamdetect_some](#) - Only show spamdetect header for bad scores
- 859. [g_spawn_log](#) - If true the spawns are logged to lib_spawn.log
- 860. [g_spf_baddns_skip](#) - If spf dns failure then allow message through (instead of giving retry error)
- 861. [g_spf_byemail](#) - Perform allow bounce confirmation via email.

- 862. [g_spf_debug_log](#) - Enable spf.log file
- 863. [g_spf_default](#) - (strict only) Default spf record if none found default 'mx/16 a ptr:%{d2} -all'
- 864. [g_spf_default_noblock](#) - (strict only) Only stamp headers if default spf record fails when no real spf header
- 865. [g_spf_dns_timeout](#) - Seconds to wait for dns lookups for spf, best not to change
- 866. [g_spf_domain](#) - Domain for SPF rewrite and allow messages (defaults to first domain on server)
- 867. [g_spf_enforce](#) - List of wildcard/domains to enforce spf for, e.g. paypal.com,*bank*
- 868. [g_spf_enforce_auto](#) - Enforce spf for commonly forged domains paypal.com,*bank*
- 869. [g_spf_enforce_local](#) - If spf fails and it's a local domain then skip grey listing and give allow message
- 870. [g_spf_header](#) - Use g_verify_mx_skip and apply to resulting ip
- 871. [g_spf_mode](#) - Do SPF check and then perform action, stamp | block | strict, action is conditional on [g_]spam_block settings
- 872. [g_spf_noallow](#) - Give hard bounce (no allow message) for spf failures for these domains & ignore friends
- 873. [g_spf_nocache](#) - Disable SPF cache
- 874. [g_spf_nogrey](#) - Skip SPF grey listing for these domains (require allow response)
- 875. [g_spf_norewrite](#) - Exceptions to rewrite rule, e.g. *@my.domain,bob@this.domain
- 876. [g_spf_required](#) - Require an spf entry for these domains
- 877. [g_spf_rev_skip](#) - Skip SPF checks if reverse ip name matches in this list, e.g. *.yahoo.com
- 878. [g_spf_rewrite](#) - Rewrite 'from' envelope in redirected mail (SRS)
- 879. [g_spf_rewrite_gateway](#) - Rewrite even if gateway rule applies
- 880. [g_spf_rewrite_relay](#) - Rewrite even if from ip is a host to relay for
- 881. [g_spf_share](#) - List of hosts to share allow ips with. Must all have same srs.secret file
- 882. [g_spf_skip](#) - Skip spf checks for these ip addresses, e.g. other mx hosts
- 883. [g_spf_skip_from](#) - Skip based on from, e.g. noreply@*paypal.com,..., Also skips RBL
- 884. [g_spf_skip_to](#) - Skip based on rcpt to, also skips RBL rules,...
- 885. [g_spf_timeout](#) - Seconds to wait for all spf lookups to finish, default 48 seconds
- 886. [g_spf_user_domain](#) - Make allow bounces use destination user domain name
- 887. [g_spf_very_strict](#) - (strict only) Only give 'allow' option for default spf rule failures not real ones
- 888. [g_spf_web_url](#) - Specify full url for spf byweb commands http://domain.name:port
- 889. [g_spflog_domains](#) - Specify which domains should get spflog entries sent to them.
- 890. [g_spflog_enable](#) - Enable this if this server is a frontend for a SurgeMail server users log into.
- 891. [g_spool_path](#) - Scan this directory for *.msg files to send as emails
- 892. [g_ssl_allow](#) - IP Wild card list to allow SSL encryption from
- 893. [g_ssl_allow_imap](#) - IP Wild card list to allow SSL encryption from for imap
- 894. [g_ssl_ciphers](#) - List permitted ciphers
- 895. [g_ssl_disable_sslv2](#) - Disable ssl 2.0 support for enhanced security, not recommended
- 896. [g_ssl_disable_tlsv1](#) - Disable tls 1.0 support, not recommended
- 897. [g_ssl_per_domain](#) - Create/use an SSL certificate for each domain
- 898. [g_ssl_require](#) - IP Wild card list to require SSL encryption from
- 899. [g_ssl_require_imap](#) - IP Wild card list to require SSL encryption from for IMAP
- 900. [g_ssl_require_login](#) - IP Wild card list to require SSL encryption for POP/IMAP
- 901. [g_ssl_require_out](#) - Other machines we only send to using SSL ip or domain
- 902. [g_ssl_require_web](#) - Require https for most web features (excluding blogs file sharing and surgeplus)
- 903. [g_ssl_sha1_sign](#) - Sign CSR with SHA1 instead of MD5 for enhanced security, beta testing
- 904. [g_ssl_try_from](#) - Try and start ssl mode if from this user, e.g. *@xyz.com
- 905. [g_ssl_try_not](#) - Skip ssl for these hosts
- 906. [g_ssl_try_out](#) - Try and start ssl mode to these hosts, may cause failures!
- 907. [g_stack](#) - For testing only, NEVER SET THIS
- 908. [g_stack_imap](#) - For testing only, NEVER SET THIS
- 909. [g_startup_delay](#) - Seconds to wait before starting surgmail
- 910. [g_store_dropped](#) - Store upto 5000 bad bounces in the dropped directory
- 911. [g_surbl](#) - SURBL Spam URI Realtime Blocklists
- 912. [g_surbl_reject](#) - Reject email with SURBL hits
- 913. [g_surbl_skip](#) - URL's to allow even if listed in surbl
- 914. [g_surbl_skip_ip](#) - Skip SURBL check if sender is from listed ip
- 915. [g_surbl_whois](#) - Also check whois info on suspect urls - not for busy servers!
- 916. [g_surgeblog](#) - Specialize SurgeMail as a Blog server
- 917. [g_surgeplus_delay_tell_upgrade](#) - Delay informing existing users about new SurgePlus versions for
- 918. [g_surgeplus_delay_tell_upgrade_exempt](#) - Users exempt from delayed new version informing
- 919. [g_surgeplus_hide_client_downloads](#) - Hide the links to download and install SurgePlus Windows client
- 920. [g_surgeplus_links](#) - Add web links to SurgePlus from other web interfaces (and vice versa) for users allowed

to use SurgePlus.

- 921. [g_surgeplus_log_level](#) - SurgePlus log level. 'none', 'info', or 'debug'. Default is 'info'
- 922. [g_surgeplus_online](#) - Enable online tracking in surgeplus
- 923. [g_surgeplus_pop_server_name](#) - Default pop server to set SurgePlus client download to connect to.
- 924. [g_surgeplus_port](#) - SurgePlus port to listen on, default is 7110
- 925. [g_surgeplus_secure_port](#) - SurgePlus SSL secure port, default is 7995
- 926. [g_surgeplus_smtp_server_name](#) - Default smtp server to set SurgePlus client download to connect to.
- 927. [g_surgeplus_web_port](#) - SurgePlus web port to listen. Default is to use HTTP webmail port
- 928. [g_surgeplus_web_url](#) - Direct SurgePlus users to access shared files at this url
- 929. [g_surgewall_redirect](#) - Allow redirect/responder for surgewall
- 930. [g_surgewall_split](#) - Split up surgewall messages, one per recipient
- 931. [g_surgeweb_backend_server](#) - Backend machine to connect to
- 932. [g_surgeweb_benchmark](#) - Log web request timing info for surgeweb benchmarking - matches ip addresses
- 933. [g_surgeweb_cache_less](#) - Reduce surgeweb caching
- 934. [g_surgeweb_debug](#) - Log surgeweb debug info - matches ip addresses or email addresses - avoid
- 935. [g_surgeweb_disable](#) - Disable access to SurgeWeb
- 936. [g_surgeweb_idle_timeout](#) - Idle timeout for surgeweb sessions (hours, default=48)
- 937. [g_surgeweb_logall](#) - For requests matching g_surgeweb_debug also leave all webio & temp files - avoid
- 938. [g_surgeweb_remember_timeout](#) - "Remember" timeout / max session length for surgeweb sessions (days, default=14)
- 939. [g_surgeweb_restrict](#) - Restrict surgeweb use to these accounts only
- 940. [g_surgeweb_work](#) - Path to Surgeweb cache/work files
- 941. [g_tarpit_badrcpt](#) - Delay rejection of bad recipients (in seconds, default 4s)
- 942. [g_tarpit_blackhole](#) - Reject the email one recipient at a time
- 943. [g_tarpit_drop](#) - Drop link and ban for 1 hour if tarpit limits exceeded
- 944. [g_tarpit_max](#) - Number of local recipients before slowing down per 30 minutes
- 945. [g_tarpit_max_remote](#) - Number of remote recipients before slowing down
- 946. [g_tarpit_retry](#) - Send retry error, 450 if tarpit limits exceeded
- 947. [g_tcp_que_len](#) - Length of listen queue for incoming connections
- 948. [g_tcp_read_timeout](#) - Timeout in 'seconds' on pop connections, do not adjust. (default 600)
- 949. [g_tellmail_ip](#) - Addresses to allow tellmail commands from (should never be *)
- 950. [g_thread_max](#) - Max threads allowed on this system (best not changed)
- 951. [g_thread_reuse2](#) - Reuse threads - fixes unix bug - not implemented
- 952. [g_thread_spinlock](#) - Spin more before sleeping when waiting for mutex
- 953. [g_timeout_try_later](#) - If timeout while waiting for message to arrive tell other end to retry
- 954. [g_timezone](#) - Places in timezone part of date string, e.g. +1200 NZT. Please leave blank!
- 955. [g_timezone_force](#) - Hours offset to local time, e.g. 5 (best left blank)
- 956. [g_to_valid](#) - Require an @ and dotted domain in all dest addresses
- 957. [g_tohost_local](#) - Authentication database tohost name entry to deliver locally (see g_proxy and g_route_by_tohost)
- 958. [g_toscan_path](#) - Path used for mime parts for virus scanner
- 959. [g_train_store](#) - Number of messages to store in each spam training directory (1000-5000)
- 960. [g_uidl_big](#) - Use random uidl if uidl not found
- 961. [g_unique_name](#) - A unique name for this server
- 962. [g_url_alias](#) - Allows translation from one url to another
- 963. [g_url_enable](#) - Enable widearea URL spam database
- 964. [g_url_host_noscan](#) - Disable the scan for url_host settings matching the domain in an incoming web request
- 965. [g_url_master](#) - Set if this is the central URL server (for netwin use only)
- 966. [g_url_master_to](#) - Central URL server email address (leave blank)
- 967. [g_url_redirect](#) - Sends http 301 redirect to tell browser resource has moved
- 968. [g_user_access](#) - User.cgi features granted to access groups
- 969. [g_user_access_default](#) - Default user.cgi features granted to users
- 970. [g_user_access_from](#) - When sending use from for useraccess rules
- 971. [g_user_alias](#) - Number of aliases accounts can create
- 972. [g_user_alias_file](#) - User aliases configuration file
- 973. [g_user_block_time](#) - Block chrisp from pop access for this time period
- 974. [g_user_blogs](#) - Number of blogs accounts can create
- 975. [g_user_cookies](#) - Enable browser cookies for user self management
- 976. [g_user_delete](#) - Let users delete themselves
- 977. [g_user_domainlist](#) - Who to show domain dropdown list to on user.cgi login page and 'user' pages

- 978. [g_user_filter_early](#) - Process user exceptions/filters before tagging message as spam
- 979. [g_user_friends_domain_log_disable](#) - Disable domain level friend.log file
- 980. [g_user_friends_log_disable](#) - Disable user level friend.log file
- 981. [g_user_list_quota](#) - Number of mailing lists users can create
- 982. [g_user_mail_view](#) - Whether an admin/manager can view/display users inbox mail
- 983. [g_user_mfilter](#) - Mfilter to run for individual user delivery, some features not supported
- 984. [g_user_pipe](#) - Pipe run on file just before delivery to user, \$USER\$ available on command line
- 985. [g_user_receive_rule](#) - Define valid source addresses for users in a group
- 986. [g_user_send_ip](#) - Block any ip address from sending too many emails
- 987. [g_user_send_max](#) - Maximum number of emails per day (requires SMTP AUTH)
- 988. [g_user_send_rule](#) - Define valid recipient addresses for users in a group (requires SMTP AUTH)
- 989. [g_user_send_warning](#) - Warn manager if any user sends more than this many messages per day, e.g. 5000
- 990. [g_user_send_white](#) - No limit for these ip addresses/users
- 991. [g_user_sms_quota](#) - Number of sms messages accounts can send
- 992. [g_user_status_send](#) - Number of days after which to send user status messages (0 = never)
- 993. [g_user_utoken_days](#) - Number of days a user self management login token is valid for
- 994. [g_user_utoken_expire](#) - Length of time a user self management login token is valid for
- 995. [g_user_utoken_idle](#) - Length of time a user self management login token may remain idle for
- 996. [g_user_virus_scan](#) - Allow virus scans for specific users instead of all users
- 997. [g_vanish_any_bounce](#) - Vanish all bounces, requires g_vanish_bad_bounces
- 998. [g_vanish_bad_bounces](#) - Vanish suspected spam bounces (requires g_received_name)
- 999. [g_vanish_relay](#) - Vanish bad bounces before relaying email too
- 1000. [g_vanish_virus_bounces](#) - Vanish suspected virus bounces (requires g_received_name)
- 1001. [g_verify_helo](#) - Verify helo name translates to same network as sending system
- 1002. [g_verify_image_hard](#) - Use extra difficult human verification image (used in blogs)
- 1003. [g_verify_mx](#) - Verify MX records contain senders IP address (see g_verify_mx_skip)
- 1004. [g_verify_mx_skip](#) - Use to define incoming mail gateway ips so the mx verify doesn't fail on them
- 1005. [g_verify_smtp](#) - Verify we can talk back to the SMTP port on incoming ip address
- 1006. [g_verify_timeout](#) - Seconds to wait for SMTP response, default is 10 seconds
- 1007. [g_vipre_enable](#) - Enable vipre scanner on windows
- 1008. [g_virus_allow_unmonitorable](#) - Allow unmonitorable content (avast antivirus)
- 1009. [g_virus_avast](#) - Enable AVAST virus scanner integration
- 1010. [g_virus_avast_hour](#) - Hour of day to update avast definitions, e.g. 9 = 9a.m.
- 1011. [g_virus_cmd](#) - Virus checker for mime parts, use \$FILE\$ in cmd
- 1012. [g_virus_cmd_codes](#) - List of return codes to bounce message, e.g. 1,2,3,4,5
- 1013. [g_virus_cmd_drop](#) - Drop silently instead of reject at data stage - not recommended
- 1014. [g_virus_cmd_email](#) - Set if scanner can understand email message files
- 1015. [g_virus_cmd_max](#) - Max concurrent threads that should run this command, if exceeded messages are not checked
- 1016. [g_virus_cmd_nodel](#) - Disables cleanup of scanned files, so you can test manually
- 1017. [g_virus_cmd_size](#) - Max size of messages to scan
- 1018. [g_virus_cmd_sleep](#) - Milli seconds to wait after g_virus_cmd incase delete is not immediate, e.g 500 = half a second
- 1019. [g_virus_disable_local](#) - Disable scanning for local trusted users
- 1020. [g_virus_disable_remote](#) - Disable virus scans for non-local addresses
- 1021. [g_virus_filter](#) - Virus checker which works like an authent module (talk to on stdin/stdout) - vpipe
- 1022. [g_virus_filter_require](#) - If any g_virus_filter pipe fails bounce messages rather than allow to continue
- 1023. [g_virus_fprot](#) - Port for FProt mail scanner (usually 11200)
- 1024. [g_virus_late](#) - Run virus scan after most spam filter processing
- 1025. [g_virus_localhost](#) - Don't skip virus checks for 127.0.0.1 originating emails
- 1026. [g_virus_recent_skip](#) - Skip virus recent cache which attempts to speed up virus scanners
- 1027. [g_virus_rename](#) - Rename executables by changing '.' to '_' prevents many auto run viruses
- 1028. [g_virus_report](#) - Report detected viruses to someone
- 1029. [g_virus_report_all](#) - Report every virus using g_virus_report
- 1030. [g_virus_restart](#) - Restart vpipe virus scanners every this many items
- 1031. [g_virus_simple](#) - Enable internal simple virus scanner
- 1032. [g_virus_simple_test](#) - Compare with avast results
- 1033. [g_vpipe_concurrent](#) - Concurrent requests to vpipe process, default is 7, set to 1 to debug vpipe issues
- 1034. [g_vpipe_fail_crash](#) - If virus scanner fails, crash surgemail (for debugging)
- 1035. [g_vpipe_notag](#) - Disable headers showing vpipe results in messages

- 1036. [g_vpipe_skip](#) - Skip scanner for this IP address (e.g. trusted mailing lists)
- 1037. [g_vpipe_timeout](#) - Timeout if scanner takes this long to respond default 60 seconds
- 1038. [g_warning_to](#) - Addresses to treat as local and send warning bounces to
- 1039. [g_web_access_grp](#) - Restrict user groups to specific ports
- 1040. [g_web_access_ip](#) - Restrict access to web ports based on ip
- 1041. [g_web_access_max](#) - Maximum number of concurrent web logins for group
- 1042. [g_web_admin_max](#) - Maximum number of concurrent web admin sessions
- 1043. [g_web_charset](#) - Charset for html pages
- 1044. [g_web_force_doctype_first_disable](#) - Disable webserver behaviour to force doctype definitions to be displayed first.
- 1045. [g_web_hide_source_names](#) - Hide the name of the source template page in output web pages.
- 1046. [g_web_max](#) - Max concurrent web connections, default is 100
- 1047. [g_web_max_perip](#) - Max concurrent web connections per-ip, default is 30
- 1048. [g_web_noserver](#) - Disable Server header in http responses
- 1049. [g_web_old_behaviour](#) - Revert to old style webserver behaviour
- 1050. [g_web_php_exe](#) - Path to php.exe
- 1051. [g_web_ref_path_extension](#) - Path extension to add to web page image/css references.
- 1052. [g_web_timeout](#) - Timeout for web requests
- 1053. [g_web_title](#) - Title to use on specified web page
- 1054. [g_web_url_path](#) - Url to path translation with access specifier
- 1055. [g_web_utf8](#) - Make sure all user.cgi handling is done in UTF8
- 1056. [g_webdav_enable](#) - Enable webdav access for users
- 1057. [g_webdav_group](#) - Only allow webdav if member of webdav access group
- 1058. [g_webdav_path](#) - Root path for webdav storage
- 1059. [g_webdav_public](#) - Enable non authenticated access to pub folder (readonly)
- 1060. [g_webmail_limit](#) - Maximum number of concurrent webmail requests
- 1061. [g_webmail_popmode](#) - Use POP3 instead of IMAP in WebMail.
- 1062. [g_webmail_port](#) - HTTP Webmail port to listen on, default is 7080
- 1063. [g_webmail_secret](#) - Secret string used by webmail when sending the ip address of connecting users
- 1064. [g_webmail_secure_port](#) - HTTPS secure WebMail port, default is https 7443
- 1065. [g_webmail_select_domain](#) - Send select_domain instead of host in webmail autologins
- 1066. [g_webmail_timeout](#) - Timeout for webmail or any cgi process (in seconds, default 360)
- 1067. [g_webmail_url](#) - Url to the WebMail cgi
- 1068. [g_webmail_urladd](#) - Url data to append to WebMail auto-login link
- 1069. [g_webmail_useip](#) - Use the ip address in g_webmail_port setting
- 1070. [g_webmail_workarea](#) - Path to WebMail workarea
- 1071. [g_work](#) - Workarea for temp files
- 1072. [g_xauthuser_hide](#) - Hide X-Authenticated-User header in processed mail
- 1073. [g_xfile_allow](#) - Allow xfile & web upload features for users. Set to '*'
- 1074. [g_xrcpt_hide](#) - Hide X-Rcpt-To header in locally delivered mail (not recommended)
- 1075. [g_xrcptoriginal_hide](#) - Hide X-Rcpt-Original header in locally delivered mail
- 1076. [g_xserver_hide](#) - Hide X-Server header in processed mail

WebMail settings

1. [see separate manual](#) - further documentation to be completed

Domain Specific Settings

Note: Most 'matching' settings take **wild card lists** as parameters, for example "fred*" will match "freddy" and "Fred@bob". And "1.2.*;2.3.*" will match 1.2.4.4 and 2.3.99.100.

address - Virtual Domain IP

This is not a setting itself but part of the vdomain setting. The vdomain setting is like a section heading, it divides the configuration file into sections, the global section comes first followed by any number of domain sections or vdomain blocks.

The address part of the vdomain setting is the IP number of this virtual domain. You will also need to configure your operating system and network to respond to this IP address. Doing this for specific operating systems is described on [this page](#) in more detail.

domain_name - Domain MX Record name

This is not a setting itself, but part of the vdomain setting. The vdomain setting is like a section heading it divides the configuration file into sections. The global section comes first followed by any number of domain sections or vdomain blocks.

This is the name of this virtual domain of the mail server. It is a domain name it accepts mail for. The mail server supports any number of virtual domains. See [this page](#) for a discussion on different types of virtual domains.

eg: if you are wanting to send mail to user@mydomain.com this setting should be "mydomain.com". But if you are wanting to send mail to user@mail.mydomain.com this setting should be "mail.mydomain.com"

This setting is separate from your actual hostname of the system running your email server. These will often be the same but if they are not it is important that the [url_host setting](#) is set to correctly resolve your server. eg: You could have domain_name = mydomain.com and url_host = mail.mydomain.com if your mail server is separate from your web server and your main company web server is already hosted on mydomain.com.

abook - Define surgeweb shared address books for this domain

The read/write fields should be a list of valid usergroups or * for all users

Syntax: abook name=string read=string write=string type=string

access_group_default - Default group to place users in

Specifies the default g_access_group to place users in this domain into.

Syntax: access_group_default string

admin_access_default - Default features granted to domain admins in this domain

This setting allows you to specify default access to certain SurgeMail features. It is specified in the same manner as the [g_admin_access](#) settings 'access' parameter. eg:

```
admin_access_default "all,!users,!reports"
```

Syntax: admin_access_default string

alias_file - Alias file

In addition each domain has its own 'alias' file (domain.name/alias.dat). You can create alias files using the same syntax as used in UNIX systems /etc/aliases. The format is:

```
username: destination
bob:      fred@domain.com
joe:      joesmith
```

This file only exists for backward compatibility.

Syntax: alias_file string

alias_max - Maximum number of aliases for this domain

Limits the total number of aliases allowed in this domain to the value specified.

Syntax: alias_max int

assume_created_epoch - If user has no 'created' field assume they were created an arbitrarily large time in the past

This setting effect the g_disable_smtp_after and g_delete_user_after settings which, by default, ignore users who have not logged in and have no created field.

Syntax: assume_created_epoch bool

blogs_max_per_user - Number of blogs each account can create

This setting has no further documentation currently available

Syntax: blogs_max_per_user int

Example: blogs_max_per_user 10

See also: [g_access_group](#), [g_blogs_max_per_user](#), [g_user_blogs](#)

broad_sync - Broadsoft Sync Enable
Customer specific feature

Syntax: broad_sync bool

centipaid - see [centipaid.htm](#)

Specifies accounts that can charge for incoming email.

Syntax: centipaid match=string acct=string pass=string https=string lang=string amount=int enabled=bool friends=bool smite=int

class - Define class of user for following commands to apply to
hidden

Syntax: class type=string from=string users=string groups=string name=string

comment - Management notes and comments about the domain

This is a dummy setting that lets you store information in the ini file that will survive setting changes from the web admin tool.

Syntax: comment date=string name=string comment=string

create_block - Block new users from this IP

Stops users from specific IP addresses from registering in this domain (assuming that you have allowed users to register themselves). Use this to stop known spammers from re-registering on your system.

Syntax: create_block string

create_cleanup - Cleanup existing data before adding a user

This causes a delete to be actioned for a user before/as they are created. This ensures the new user does not end up with any files, on any mailing lists, with any aliases etc from a previous user of the same name/address. If you delete users from the authent database directly i.e. not using the surgemail web admin or calling 'tellmail delete_user' then this setting will cleanup the users files when their address is re-used.

Syntax: create_cleanup bool

create_delete_days - Number of days a disabled new account remains before deletion

Accounts disabled with create_disable_days remain until the specified number of days at which point they are deleted.

Syntax: create_delete_days int

See also: [create_disable_days](#)

create_disable_days - Number of days new accounts remain active for

New accounts when created are set to expire after the specified number of days. When this occurs they can no longer login or receive email.

Syntax: create_disable_days int

See also: [create_delete_days](#)

create_image - Use verification image on signups

This adds a verification image to the signup process. The user will be required to correctly enter the value shown in the image to signup

Syntax: create_image bool

create_linkto - Link to redirect to after successful live account creation

This is the web url/link that the user creation process links to once it has created a new and active email account (active means it is ready for use, i.e. not disabled, verified my manager etc)

Syntax: create_linkto string

create_max - Maximum signups from one IP in a day

This setting stops spammers registering hundreds of accounts on your system before sending out a lot of spam. A setting of 2-3 is probably a good idea.

Syntax: create_max int

create_repass - User enters password twice on creation

If true this will show a "Password Again" input in addition to the "Password" input on the user signup page. The user is required to type the password in twice and the passwords are compared to ensure they are identical.

Syntax: create_repass bool

create_reqd - Required fields for new users

A comma separated list of field names. Allowable field names are the "field" value(s) of the g_authent_info setting.

For example, if your setting is:

name,phone

then when a new user is created they will be forced to fill in the name and phone fields in the registration form.

NOTE: A g_authent_info setting is required to make the field appear on the signup page, eg.

g_authent_info name="Phone" field="phone" access="user" default=""

the above setting causes a field for phone to appear on the signup page and in the user details page.

Syntax: create_reqd string

create_subdomain - Allows users to create an account that belongs to it's own unique domain.

If true this allows users to create accounts with a unique subdomain i.e. firstname@lastname.domain. SurgeMail uses the [domuser system](#) to handle sub-domain users. Wildcard [MX records](#) are required to ensure delivery to subdomain users.

Syntax: create_subdomain bool

create_tpl_dir - Relative path to user create pages

The relative path from the web directory to the user creation and user self management pages, these pages are typically called na_*.htm and stored in the /web directory. If you want a different look and feel for a domain simply set this and copy the pages to a directory in /web then modify them.

For example, if your setting is:
otherdomain/

Then you would:
cd /surgemail/web
mkdir otherdomain
copy na_*.htm otherdomain
CD otherdomain
notepad na_login.htm

Syntax: create_tpl_dir string

create_user - Method for adding new users

Can be one of:

Value	Description
open	Anyone can create an account immediately providing name and password. Be sure to set create_max "3" to prevent spammers creating dozens of accounts.
email	Anyone can create an account providing existing email address.
manager	Manager approves account, user provides existing email address.
manager_new	Manager activates account, user proves name and password.
disabled	Users cannot signup, accounts are created by the Web Admin.

In 'open' mode users account are created instantly. In 'email' mode they recieve an email and use a link to create their account. In 'manager' mode the manager recieves an email and via a link sends the user an email which they use to create their account. In 'manager_new' mode the account is created disable and the manager receives an email with a link to activate the account.

Syntax: create_user string

delete_user_after - Number of days an account can remain unread before it is deleted

DO NOT USE THIS SETTING IN A MIRROR/CLUSTER SETUP

Number of days an account can remain unread before it is deleted. This setting cannot be used on an authent_domain FALSE domain unless it has a [prefix](#) setting.

Syntax: delete_user_after int

disable_smtp_after - Number of days an account can remain unread before delivery is disabled
Specifies a number of days an account can remain 'unread' before it stops recieving new emails. This is intended to stop mail piling up for abandoned email accounts.

Syntax: disable_smtp_after int

disable_surgeplus - Disable SurgePlus Calendar and File Sharing client

Disable users from logging in using the SurgePlus Calendar and File Sharing client. See [SurgePlus](#)

Syntax: disable_surgeplus bool

See also: [g_disable_surgeplus](#)

dmail_bin_path - Path for DMail bin files to automatically convert

Path for DMail style bin files to automatically convert. This allows you to import delivered but unpopped mail from DMail bin and mailbox files. While this is set a check is done whether this import needs to be done each time a user logs on. Any mail is converted on the fly and added to the users SurgeMail inbox.

Syntax: dmail_bin_path string

dmail_deliver - Deliver messages into dmail drop directories (not supported)

This setting has no further documentation currently available

Syntax: dmail_deliver bool

dmail_drop_path - Path for drop files to automatically convert

Path for DMail / sendmail style drop files to automatically convert. This allows you to import delivered but unpopped mail from standard drop files. While this is set a check is done whether a drop file needs to be imported each time a user logs on. Any mail is converted on the fly and added to the users SurgeMail inbox.

Syntax: dmail_drop_path string

dmail_drop_prefix - Whether prefix is used on dmail drop files

Prefix on dmail drop files for dmail_drop_path conversion.

Syntax: dmail_drop_prefix bool

dmail_hash - Hashing scheme used by dmail_drop_path and dmail_bin_path

DMail style hashing scheme used by dmail_drop_path and dmail_bin_path.

Syntax: dmail_hash string

dmail_skip_imap - Skip conversion of old imap *.mbx folders

This setting is usually not needed

Syntax: dmail_skip_imap bool

encrypt_limit - Max encrypted msgs per user per hour

Per user limit

Syntax: encrypt_limit int

encrypt_noconfirm - Disable confirmation for encrypted messages

Disables the automatic confirm reading message

Syntax: encrypt_noconfirm bool

encrypt_rule - Matches will be encrypted when sent

If this rule matches then the message will be encrypted before it is sent to the user. method=server or inline, we recommend 'server' mode as it's much simpler.

Syntax: encrypt_rule header=string contains=string from=string to=string noconfirm=bool method=string

encrypt_smart - Encrypt smart features enabled for this domain

Not for general use

Syntax: encrypt_smart bool

encrypt_subject - Subject when encrypted message sent - default is original subject

Not yet implemented

Syntax: encrypt_subject string

encrypt_token - Send token to 'sender' for new SurgeVault recipients

Not for general use

Syntax: encrypt_token bool

enotify_from - From address to use in email notification messages

Users can set an email address to send a notify to when they get an email. This setting sets the 'from' header for such messages.

Syntax: `enotify_from` string

See also: [g_enotify_from](#)

expire_age - Expire undeleted mail older than specified age left in INBOX

Use this to trim messages that are left in the INBOX, this means messages that are unread and messages which are read but not moved to another folder. The deleted messages are replaced with a single message explaining which items have been deleted (with from, subject and date of each message deleted). You should define both age and size to enable expiration.

Currently expire ONLY affects 'newmail' ie: mail that is waiting to be read not mail stored in IMAP folders.

Syntax: `expire_age` int

expire_rule - Expire rules for specific folders

These rules let you specify and expire rule for any folder. The folder match is not case sensitive

e.g. to delete all spam message over 30 days old and larger than 3k and to empty the trash can each day.

```
expire_rule folder="Spam" age="30" size="3k"
```

```
expire_rule folder="Trash" age="1"
```

You can use wild cards, e.g. Delete all messages over 90 days old larger than 10k unless in the new or archive* folders.

```
expire_rule folder="*,!new,!archive*" age="90" size="10k"
```

Syntax: `expire_rule` folder=string age=int size=int

expire_size - Expire undeleted mail larger than specified size (units=bytes)

Use this to trim messages that are not read by users. The deleted messages are replaced with a single message explaining which items have been deleted (with from, subject and date of each message deleted so that anything really important can be recovered). You should define both age and size to enable expiration.

Syntax: `expire_size` int

fallback - Fallback Email address or account

Specifies a default account to deliver Email to. This is sent to a non existent account. If not defined the Email will be bounced. Setting fallback to "/dev/null" will drop messages (both UNIX and Windows).

Syntax: `fallback` string

fallback_always - Also relay to old system even if user does exist - not recommended

This setting can be used when bringing up a new system if you want to be able to backout. It is not recommended

Syntax: `fallback_always` bool

fallback_check - Fallback check

Check if user exists on fallback relay host before accepting it.

Syntax: `fallback_check` bool

fallback_relay - Fallback host to relay non existent accounts to

Specifies a default host to send messages to that are not found in the local user database. This allows you to transition between two mail systems, as new accounts are created the emails will be delivered to SurgeMail, and ones that don't exist will be sent on to the old system automatically. There are [several options](#) to make this work using servers that only accept mail if they can do a reverse lookup.

Syntax: `fallback_relay` string

fallback_users - Path to file listing all users to user fallback_relay for

Useful to remove load from backend server as it doesn't have to be checked for non existent users, the file can contain user@domain or just usernames

Syntax: fallback_users string

footer_file - Footer file for plain text messages

Footer file for all plain text messages 'from' this domain based on from address.

Syntax: footer_file string

footer_html - Text footer file

Footer file for all HTML messages 'from' this domain based on from address.

Syntax: footer_html string

forward_illegal - Prevents users setting forward rules to certain addresses

Syntax: g_forward_illegal to="address" apply="user type "

This setting allows you to specify some addresses as being illegal for certain users. This stops users setting up forwarding rules to these addresses. They can still send mail to these addresses manually with their email client. These rules **_ONLY_** apply to non local domains.

Some examples:

If you want to stop your users setting up forward rules that redirect to aol.com.

g_forward_illegal to="*@aol.com" apply="user"

If you want to stop your users setting a forward to all domains except aol.com

g_forward_illegal to="*,!*@aol.com" apply="user"

Stop domain admins sending to aol.com

g_forward_illegal to="*@aol.com" apply="domadmin"

Stop admins sending to netwinsite.com

g_forward_illegal to="*@netwinsite.com" apply="admin"

Syntax: forward_illegal to=string apply=string

friends_at_rcpt - Whether to check users friends list at rcpt stage

This setting is automatically added/removed by the web admin when domain level friends defaults are configured. It allows us to check friends at rcpt stage without paying a disk access cost for non-friends users.

Syntax: friends_at_rcpt bool

friends_pending_name - The imap name of the friends_pending folder default is 'Friends Pending'

This shouldn't be changed unless this feature has not been used before as it will confuse your users as they will have the old imap folder too. Ensure you have enabled this feature with g_imap_friends setting

Syntax: friends_pending_name string

See also: [friends_at_rcpt](#), [friends_url](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_bounce_friend](#), [g_friends_daemon_ok](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_lang_auto](#), [g_friends_pending_keep](#), [g_friends_pending_max](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_safer](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_global_add](#), [g_friends_global_exclude](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_long](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_byemail](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_friends_debug1](#),

[g_imap_friends](#), [g_quota_friends](#), [g_spam_probe_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

friends_url - Specify full url for friends release <http://domain.name:port> domain specific setting
Normally the default will work.

Syntax: friends_url string

from_exact - Check from matches authenticated user
Checks that the from header perfectly matches the authenticated user

Syntax: from_exact bool

gateway_to - Send all email to another server
Useful when migrating if you want email to go to another server while users can still login and read existing email on this server, the messages will be sent even if the user does not exist locally

Syntax: gateway_to string

header_add - Add header to posts 'from' this domain
Adds headers, the headers are added based on the 'from' domain of the message.

Syntax: header_add string

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_from_rewrite_sender](#), [g_footer_send](#), [g_footer_sendonly](#), [g_responder_sender](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_send_store_disable](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

host_alias - Alias name(s) for this virtual domain

When a user sends to 'bob@xx.your.domain.name' or 'bob@yy.your.domain.name' you need to have the alias host names 'xx.your.domain.name' etc, defined or the mail server will reject the message. Wild card's can be used for this setting. Example:
host_alias "*.your.domain.name" This can also be used to accept mail directly to the servers ip address eg 'bob@123.4.5.'

Syntax: host_alias string

imap_public - Share IMAP folders between users

This setting allows folders to be shared between users. See [g_imap_acl](#), Requires surgemail 3.9d or later!.
e.g. `imap_public name="INBOX" alias="lances_inbox" user="lance" users="*"`

Syntax: imap_public name=string alias=string user=string subfolder=bool users=string group=string
readonly=bool

Example: Share IMAP folders between users

See also: [dmail_skip_imap](#), [imap_public_show](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_debug_imap](#), [g_fix_imap_if](#), [g_imap_acl](#), [g_imap_auto_create](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_debug](#), [g_imap_idle_free](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_throttle](#), [g_imap_timezone](#), [g_imap_timeout](#),

[g_imap_uidl_nofix](#), [g_imap_unsub_auto](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#),
[g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#),
[g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#),
[g_stack_imap](#)

imap_public_show - Auto subscribe public folders

This setting has no further documentation currently available

Syntax: `imap_public_show bool`

language_default - Default language for user web interface

If the user has not yet selected a language then this language is used as a default. If the language specified here does not exist in the language files, or nothing is specified here then English is used as the default language.

Syntax: `language_default string`

late_forward - Apply domain users forwarding rules after friends, spam, and filtering

By default users forwarding rules are applied before friends, spam and user filter rules. By default users can tick an option on their forwarding page to perform 'late' forwarding, that is forwarding that occurs after friends, spam and filtering. This option overrides the user option and causes domain users forwarding rules to be applied after friends, spam and filtering.

Syntax: `late_forward bool`

ldap_anydomain - Lets users search other than their own domain in ldap

If ldap is enabled for the system (`g_ldap_port`) then this lets users of this domain lookup users in any domain not just this domain.

Syntax: `ldap_anydomain bool`

See also: [ldap_disable](#), [g_ldap_port](#), [g_ldap_forward](#), [g_ldap_outlook_browse_max](#)

ldap_disable - Stops ldap logins by users of this domain

If ldap is enabled for this system, then this setting disables it for this specific domain

Syntax: `ldap_disable bool`

See also: [ldap_anydomain](#), [g_ldap_port](#), [g_ldap_forward](#), [g_ldap_outlook_browse_max](#)

legal_archive_disable - Disable legal archive for this domain (experimental)

This setting has no further documentation currently available

Syntax: `legal_archive_disable bool`

legal_archive_hide - Hide legal archive for this domain (experimental)

This setting has no further documentation currently available

Syntax: `legal_archive_hide bool`

legal_archive_keep - Days to keep legal archive, units=days unless you specify years or months default is `g_legal_archive_keep`

Syntax: `legal_archive_keep int`

list_disable - Disables creation of mailing lists

When set to "TRUE" this disables mailing list creation for this domain.

Syntax: `list_disable bool`

list_max - Maximum number of mailing lists for this domain

Set this to the maximum number of mailing lists to allow for this domain.

Syntax: list_max int

list_max_users - Maximum number of users allowed in all lists in this domain

This is a quota of users/members for all lists in this domain. The maximum number of members in each list in this domain must total to less or equal to this setting.

eg:
list_max_users "100"
list_max "2"

In this scenario, 100 users could be used in 2 lists. So one list might have 80 users the other 20, but the combined total must be less than or equal to 100 users.

Syntax: list_max_users int

loginfails - Disconnect after failed logins

Disconnect user after this many bad password guesses.

Syntax: loginfails int

lookup_relay_on_from - Lookup from addresses for relay allowed

Looks up local authenticated smtp from addresses to check for relay is allowed flag (relay="true").

Syntax: lookup_relay_on_from bool

mailbox_path - Path to mailbox maildir (inbox) files

Specifies the root directory for users in this domain for their incoming mail messages and mail folders (for IMAP), maildir structure is used and hashing will also be applied so if you specify d:\spool, then 'bob's Email will appear in d:\spool\xx\yy\bob\mdir... where 'xx' and 'yy' are hashing numbers for that user. (Hashing is required to keep directory performance at a high level when you have millions of users).

Syntax: mailbox_path string

manager_email - Managers Email

This is the manager's Email address for this domain. When users register themselves, if you have set create_user to the 'manager' method, an Email will be sent to this Email address to await confirmation of the user creation.

Syntax: manager_email string

manager_username - Domain managers username (for web based domain administration)

Specifies the local users which have manager rights for this domain. These users can login to the user self management interface and will receive special domain manager options. If you specify an account without the @domain part i.e. 'admin' it assumes admin@.

Syntax: manager_username string

msg_max_in - Max size of incoming messages for this domain.

Sets the size of messages for this domain, note that this may affect the ehlo response but only for 'address' based virtual domains, so you must ensure your g_msg_max setting is sufficiently large. Also since this figure may be shown before the msg_max_out value is determined you must also make it larger than the msg_max_out value. We don't recommend using this setting unless it is totally necessary. It is better to choose a g_msg_max setting that all domains can live with.

Syntax: msg_max_in int

msg_max_out - Max size of outgoing messages for smtp authenticated users

Sets the size of messages for this domain, as the email client may have already seen the value of `g_msg_max` or `msg_max_in` it may not help setting this value 'larger' than those other values. **We do not recommend using this setting**, it is best to choose a `g_msg_max` value that all domains can live with.

Syntax: `msg_max_out` int

old_imaphost - Intercept mode migration for IMAP folders servers

The `old_pophost` settings will create a local account and download many mail in your inbox. However in the event that your old server also was an IMAP server you will be able to migrate your stored message folders using the `old_imaphost` setting. This download is only ever attempted once and does so asynchronously. A 300MB mailbox with 15000 messages will would be expected to take around 20 minutes. While IMAP folders are being downloaded the mailbox can already be used. Note: the mail on the old server gets deleted.

Upon IMAP login an `old_pophost` check is also performed if defined. This is specifically so that WebMail accessing SurgeMail using IMAP (recommended configuration) will allow the retrieval of mail from old POP servers.

Syntax: `old_imaphost` string

old_imaphost_always - Always attempt to suck mail on each IMAP login (slow)

This setting will force the download of mail and folders from the old server upon each IMAP login. Note: that this should only be used if specifically required as this will happen for example each time that a WebMail change made.

Note: This will obviously stop retrieving mail if the user changes their password in SurgeMail but not on the old server.

Syntax: `old_imaphost_always` bool

old_imaphost_createuser_disable - Disable user creation in database on login

If you have already got your users in your authentication database and do not wish to add new users logging in using intercept mode this setting can be used to prevent user creation upon first login to SurgeMail using POP.

Syntax: `old_imaphost_createuser_disable` bool

old_imaphost_file - Migration based on file

Specialist setting for one specific system, not for general use.

Syntax: `old_imaphost_file` string

old_imaphost_lowercase - Lowercase all migrated folders

Lowercase all migrated folders.

Syntax: `old_imaphost_lowercase` bool

old_imaphost_nodelete - Leave mail on the old server

This setting will leave mail on the old server just in case there are problems with the migration. Note: the use of this setting will disable the use of `old_imaphost_always`.

Syntax: `old_imaphost_nodelete` bool

old_imaphost_nodomain - Strip domain when logging in to old_imaphost

This can be used if you are migrating from a server that uses username only (without domain) logins.

Syntax: `old_imaphost_nodomain` bool

old_imaphost_file, old_imaphost_user, old_imaphost_pass - Migration based on file settings

Specialist file/IMAP migration based on file settings.

Syntax: `old_imaphost_pass` string

old_imaphost_prefix - Mail prefix for old imap server when using old_imaphost

IMAP prefix for old imap server. eg. mail/.

Syntax: old_imaphost_prefix string

old_imaphost_skip - Skip folders

Comma separate wild card list of migrate folders to skip past.

Syntax: old_imaphost_skip string

old_imaphost_user - Migration based on file - user field **Specialist setting for one specific system, not for general use.**

Syntax: old_imaphost_user string

old_pophost - Old pop host for pop intercept mode based migration

Specifies an old POP host that can be used when [migrating](#) users from an old mailserver to a new mailserver. This will create a local accounts with a identical username/passwords and retrieve all mail from the old server for the old account when the user logs into SurgeMail for the first time and they are not yet in the SurgeMail user database. Mail on the old server is deleted.

The use of old_pophost adds an additional check (based on partial rcpt delivery - see g_badfrom_check) to user account self creation to prevent user creating accounts that clash with existing accounts that have not been popped on SurgeMail. This means that the old server should only accept delivery to actual accounts or all user account self creation will be disabled.

Syntax: old_pophost string

old_pophost_always - Always attempt to suck mail on each login

Suck mail from old_pophost on each login (account information is not set at each login). This allows the user to be using the new mail server and still retrieve mail from the old server if mail is delivered there. This is useful in two cases:

- 1) if the user is already using SurgeMail but some mail is still delivered to the old server due to delays in MX record propagation.
- 2) To allow incremental migration. Some users can be using the SurgeMail and some users can be using the old server. Users still on the old server sending mail to users on SurgeMail would deliver to the old server. When a user on SurgeMail logs into SurgeMail any such mail is retrieved from the old server.

Note: This will obviously stop retrieving mail if the user changes their password in SurgeMail but not on the old server.

Syntax: old_pophost_always bool

old_pophost_bind - Bind outgoing connection during pop migration **This binds to the specified local port during a migration**

Syntax: old_pophost_bind string

old_pophost_createuser_disable - Disable user creation in database on login

If you have already got your user in your authentication database and do not wish to setup this setting can be used to prevent user creation upon first login to SurgeMail using POP.

Syntax: old_pophost_createuser_disable bool

old_pophost_nodelete - Leave mail on the old server

This setting will leave mail on the old server just in case there are problems with the migration. Note: the use of this setting will disable the use of old_pophost_always.

Syntax: old_pophost_nodelete bool

old_pophost_nodomain - Strip domain when logging in to old_pophost

This can be used if you are migrating from a server that uses username only (without domain) logins.

Syntax: old_pophost_nodomain bool

old_pophost_sep - Login separator

Seperator, default is '@'. e.g. some systems use %

Note: The old_pophost_iffirst and old_pophost_makeuser have now been replaced by the more consistent old_pophost_createuser_disable setting.

Syntax: old_pophost_sep string

old_smtp host - Old smtp host

SMTP host to check for existing users (when creating new accounts).

Syntax: old_smtp host string

old_smtp host_skip - Skip old_smtp host checks for administrators

Who to skip old_smtp host checks for. Valid values are "admin" and "domadmin".

Syntax: old_smtp host_skip string

old_xfile - Migration - Copy xfile data across

Only valid if the old server is also running surgemail. And requires old_pophost settings to work

Syntax: old_xfile bool

pop_min_time - Min seconds between pop logins (see warning)

This setting confuses some clients, and results in the user having to login again. The error is not always shown to the end user by some dumb email clients

Syntax: pop_min_time int

pop_welcome - Welcome message for POP/IMAP

This is the string displayed to the user when they connect to this domain before they login. The same string is also used in IMAP response. See also smtp_welcome.

Syntax: pop_welcome string

prefix - Prefix for usernames in database

This prefix is used in the user database to distinguish these virtual domain users. This setting is for backward compatibility and not generally recommended. It is better to store user@domain.name in the userdatabase rather than just 'username'.

Syntax: prefix string

proxy_pop_nodomain - Strip domain when talking to proxy POP host

This setting causes the domain name to be stripped from user login names when talking to the proxy POP host. This does not apply to [surgewall](#), see [surgewall_options](#) for details

Syntax: proxy_pop_nodomain bool

quota_default - Default Email quota for users

This setting allows you to limit disk usage of each user a setting of 10mb is typical. The main reason for this setting is to stop a single user who is being mail bombed using up all your disk space. So even if you don't want to limit disk use you should still set some limit eg:100,000,000 (100mb)

Syntax: quota_default string

quota_domain - Total quota for the domain, e.g. 300mb, 2gig

Limits the total usage (used quota) for the entire domain. Note that the command tellmail quota_rebuild_domain domain.name may be used to reset these figures.

Syntax: quota_domain string

See also: [quota_default](#), [user_sms_quota](#), [user_list_quota](#), [webdav_quota](#), [g_log_quota](#),

[g_quota_warning_disable](#), [g_quota_noemail](#), [g_quota_rcpt_disable](#), [g_quota_try_later](#), [g_quota_friends](#), [g_quota_skip](#), [g_quota](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

rcpt_msg - Response given for invalid recipient errors, message is prefixed by email address.

This setting has no further documentation currently available

Syntax: rcpt_msg string

See also: [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

redirect - Redirect Email to another account

This redirects mail from one user to another. The destination can be a full Email address with another domain name.

Syntax: redirect was=string to=string

redirect_cc - CC & Redirect Email to another account

This carbon copy redirects a message so the original user receives it as well as the new user you have specified. This is good for keeping a record of incoming emails for a particular account.

Syntax: redirect_cc was=string to=string

redirect_hash - Share incoming message evenly between several accounts

The sharing is done based on a hash of the 'from' address so that the same 'from' address will always go to the same recipient

Syntax: redirect_hash was=string to=string

redirect_max - Limits the number of redirect rules

This setting applies a limit to the number of redirect rules which are allowed in this domain (only applies to domain admins)

Syntax: redirect_max int

security_suffix - Suffix for smtp/imap/pop login

This setting stops the username matching the email address by requiring a different suffix for logging in, so user@xyz.mail.server instead of user@mail.server. This is like an alias for the domain that only works for logging in.

Syntax: security_suffix string

See also: [disable_smtp_after](#), [old_smtphost](#), [old_smtphost_skip](#), [smtp_auth_off](#), [smtp_welcome](#), [smtp_welcome_name](#), [smtp_from_ip](#), [surgeweb_backend_smtp](#), [surgeplus_smtp_server_name](#), [g_disable_smtp_after](#), [g_dbabble_smtp_port](#), [g_dbabble_smtp_prefix](#), [g_deny_smtp](#), [g_safe_smtp](#), [g_safe_smtp_email](#), [g_manager_smtp](#), [g_smtp_auth_debug](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_help_disable](#), [g_smtp_plain_hide](#), [g_smtp_cram_enable](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#), [g_smtp_thread](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauth](#), [g_smtp_noauthm](#), [g_smtp_noauth_msg](#), [g_verify_smtp](#), [g_surgeplus_smtp_server_name](#)

send_helo - Mail host A Record name used when sending helo to other servers - requires

[g_send_helo_from](#) true

This is only used if [g_send_helo_from](#) is also true

Syntax: send_helo string

smtp_auth_off - Disable SMTP AUTH from unknown ip addresses

This prevents a hacker sending out spam by cracking a users account details, users must login from an address specified in g_smtp_auth_ip or g_relay_allow_ip

Syntax: smtp_auth_off bool

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surge web backend smtp](#), [surge plus smtp server name](#), [g disable smtp after](#), [g dbabble smtp port](#), [g dbabble smtp prefix](#), [g deny smtp](#), [g safe smtp](#), [g safe smtp email](#), [g manager smtp](#), [g smtp auth debug](#), [g smtp bounce nslow](#), [g smtp cmd timeout](#), [g smtp data timeout](#), [g smtp delay stamp](#), [g smtp welcome delay](#), [g smtp log protocol](#), [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nlimit](#), [g smtp max bad](#), [g smtp port](#), [g smtp port auth](#), [g smtp port force](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp help disable](#), [g smtp plain hide](#), [g smtp cram enable](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix no head](#), [g smtp thread](#), [g smtp auth off](#), [g smtp auth ip](#), [g smtp noauth](#), [g smtp noauth m](#), [g smtp noauth msg](#), [g verify smtp](#), [g surge plus smtp server name](#)

smtp_from_ip - Require incoming email from matching ip

This is used to ensure all incoming email comes direct from a filter system and not from the internet, only authenticated email will bypass this (and g_spam_allow)

Syntax: smtp_from_ip string

smtp_welcome - Welcome message for SMTP

This is the string displayed to the user when they connect to this domain, before they login. See also pop_welcome

Syntax: smtp_welcome string

smtp_welcome_name - SMTP welcome connection hostname

This setting has no further documentation currently available

Syntax: smtp_welcome_name string

spam_block - Default for this domain to block spf etc failures

This setting sets the default behavior for this domain, if g_spam_block is not set, then this setting can turn on blocking as the default for this entire domain. Individual users can still set their own settings to block or not block for spf.

Syntax: spam_block bool

See also: [g friends allow spf](#), [g friends spf fail bounce](#), [g friends check spf](#), [g received skip spf](#), [g spf mode](#), [g spf nocache](#), [g spf rewrite](#), [g spf rewrite relay](#), [g spf rewrite gateway](#), [g spf norewrite](#), [g spf dns timeout](#), [g spf timeout](#), [g spf domain](#), [g spf user domain](#), [g spf very strict](#), [g spf debug log](#), [g spf default](#), [g spf default noblock](#), [g spf skip](#), [g spf skip from](#), [g spf skip to](#), [g spf rev skip](#), [g spf share](#), [g spf header](#), [g spf baddns skip](#), [g spf nogrey](#), [g spf noallow](#), [g spf enforce](#), [g spf enforce auto](#), [g spf required](#), [g spf enforce local](#), [g spflog enable](#), [g spflog domains](#), [g spf byemail](#), [g spf web url](#)

spam_noblock - Disable spf blocking for this domain

This is the opposite of spam_block, this can be used if g_spam_block is true and you wish to disable spf blocking for one domain.

Syntax: spam_noblock bool

See also: [g friends allow spf](#), [g friends spf fail bounce](#), [g friends check spf](#), [g received skip spf](#),

[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_rewrite_gateway](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_enforce](#), [g_spf_enforce_auto](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byemail](#), [g_spf_web_url](#)

spam_strip - Strip spam headers for this domain

Strip spamdetect headers for this domain.

Syntax: spam_strip bool

ssl_allow - IP Wild card list to allow SSL encryption from

This setting remove the capability so clients won't attempt ssl, it is only really functional for ip based virtual domains

Syntax: ssl_allow string

ssl_pop_domain - Domain to use for ssl certificates for POP and IMAP

If you have multiple aliases for this domain then this setting lets you choose which one to use for the SSL certificate

Syntax: ssl_pop_domain string

See also: [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossl](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_require_web](#), [g_ssl_try_out](#), [g_ssl_try_not](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_tlsv1](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

surgeplus_pop_server_name - Default POP server for SurgePlus clients

New installs of the SurgePlus client will be automatically configured to use this specified POP server. If you don't specify a value for this setting, then the POP server will default to what you have specified by the url_host setting, or the domain name if you don't specify a url_host setting. You will need to do a 'tellmail surgeplus rebuild' command after changing this setting and if you are downloading the new build via your web browser be aware that web browsers sometimes cache the old download.

Syntax: surgeplus_pop_server_name string

Example: pop.your.domain.name

See also: [url_host](#), [disable_surgeplus](#), [surgeplus_smtp_server_name](#), [g_url_host_noscan](#), [g_surgeplus_smtp_server_name](#)

surgeplus_smtp_server_name - Default SMTP server for SurgePlus clients

New installs of the SurgePlus client will be automatically configured to use this specified SMTP server. If you don't specify a value for this setting, then the POP server will default to what you have specified by the url_host setting, or the domain name if you don't specify a url_host setting. You will need to do a 'tellmail surgeplus rebuild' command after changing this setting and if you are downloading the new build via your web browser be aware that web browsers sometimes cache the old download.

Syntax: surgeplus_smtp_server_name string

Example: pop.your.domain.name

See also: [url_host](#), [disable_surgeplus](#), [surgeplus_pop_server_name](#), [g_url_host_noscan](#), [g_surgeplus_pop_server_name](#)

SurgeWall - Surgewall mailproxy feature (Version 1.4a or greater required)

This allows SurgeMail to be placed as a "filter" in front of an existing mailserver to apply friends rules, spam filtering and/or virus scanning. All you need to do is set this to the existing server address. POP3 will be routed through to the existing server and users can login to the SurgeMail web interface to configure their friends, spam and virus options eg:

```
surgewall "1.2.3.4"
```

This setting should be an IP address not an IP and port. Use `surgewall_options` if you need to specify non-standard ports or a different IP for POP, SMTP and/or IMAP. You may specify a comma seperated list of IP addresses. SurgeWall will connect to each in turn until it gets a successful login. To modify this behaviour see the `proxy_failover` option of the [surgewall_options](#) setting.

See [here](#) for more details.

Syntax: `surgewall` string

surgewall_auth - SurgeWall SMTP authentication
Defines the username and password to use for SMTP authentication when sending to SurgeWall'ed server.
Syntax: `surgewall_auth` user=string pass=string

surgewall_capa_local - Just return local imap capa response rather than remote
Note that it can only guess the right imap host if you are using ip based virtual domains.
Syntax: `surgewall_capa_local` bool

surgewall_local_too - For web domain admin try local database too
Allows local user accounts to be created for domain admin
Syntax: `surgewall_local_too` bool

surgewall_options - SurgeWall miscellaneous options (Version 1.4c or greater required)
This setting controls the SurgeWall miscellaneous configuration options it has several parameters:

strip_domain	TRUE/FALSE strips the domain name from the username when sending to the original server.
proxy_failover	TRUE/FALSE failover mode for several addresses, only use next address if previous one fails to respond.
auth_local	TRUE/FALSE requires that users exist locally, no authentication is done via the original server.
pop	Comma seperated list of IP addresses and port of the original POP3 server.
imap	Comma seperated list of IP addresses and port of the original IMAP server.
smtp	Comma seperated list of IP addresses and port of the original SMTP server.
usercgi	pop/imap which protocol to use when authenticating logins to the web interface.

The POP, SMTP and IMAP options allow you to configure SurgeWall to connect to different IPs and/or ports for each interface that it proxies. So for example you can run SurgeWall on the same machine as the old mail server provided the old mail server is configured to run on non-standard ports. eg:

```
surgewall_options strip_domain="TRUE" pop="127.0.0.1:111" smtp="127.0.0.1:26" imap="127.0.0.1:144"
```

or perhaps you have the pop, smtp and imap components of the server running on several machines, eg.

```
surgewall_options strip_domain="TRUE" pop="1.2.3.4:111" smtp="2.3.4.5:26" imap="3.4.5.6:144"
```

You may specify several different IPs in a comma seperated list in the POP, SMTP and IMAP options if you do this SurgeWall will connect to each in turn until it gets a successful login. The same is true for the [SurgeWall](#) setting.

To modify this behaviour you can set `proxy_failover` to TRUE this causes SurgeWall to only use the next address if it fails to connect to the preceeding address, meaning it will use each server specified only if the previous server is not responding.

Syntax: `surgewall_options` strip_domain=bool proxy_failover=bool auth_local=bool pop=string smtp=string imap=string usercgi=string

surgeweb_backend_server - Backend server to connect to

This specifies the backend machine where Surgeweb connects for email and to store user settings. Surgeweb will cache data here but store the master copy of anything on the backend machine.

Syntax: surgeweb_backend_server string

**surgeweb_backend_smtp - Backend smtp access (if non default)
per default connects to 127.0.0.1:25**

Syntax: surgeweb_backend_smtp string

**surgeweb_backend_web - Backend web access - for usercgi /surgeplus (if non default)
If no backendserver specified is '/', default with backend is 'http://backend_server/'. For non default port / machine specify say: 'server.name:7080'**

Syntax: surgeweb_backend_web string

surgeweb_custom - Surgeweb customisation level

This setting has no further documentation currently available

Syntax: surgeweb_custom string

suspend - Disable logins for entire domain

Use this where a domain is not being paid for and you want to suspend all users in the domain. This prevents users checking mail NOT users sending mail or other domains sending to this domain

Syntax: suspend bool

url_alias - Allows translation from one URL to another

Allows translation from one URL or beginning of a URL to another. eg:

url_alias from="/cgi-bin/" to="/scripts/"

will cause the URL http://localhost:7025/cgi-bin/fred.cgi to reference the same file as http://localhost:7025/scripts/fred.cgi would have, the fred.cgi in the SurgeMail 'scripts' directory. These settings are checked before the [g_url_alias](#) settings, the first matching rule is used, settings are checked in the order specified.

Syntax: url_alias from=string to=string ports=string

url_blogs - BLOGS host A Record name (if different from MX Record name - eg. blogs.mydomain.com)

This is used when generating the 'view' link, if you don't specify a port (:nnn) then it will use the first webmail port by default, by default the url_host setting will be used, failing that the domain name is used

Syntax: url_blogs string

url_host - Mail host A Record name (if different from MX Record name)

This name is used in URLs to this domain. It is important that the hostname specified here will resolve to this physical host running SurgeMail at all times.

It is used by WebMail in the email sent to user upon signup. The email sent to the manager when a user signs up and is the host passed to WebMail when auto-logging a user in. If your auto-logins are failing because of a "cannot connect error" then you may need to set this to the correct host.

Syntax: url_host string

user_access_default - Default user features granted to users in this domain

This setting allows you to specify default access to certain SurgeMail features. It is specified in the same maner as the [g_user_access](#) settings 'access' parameter. eg:

user_access_default "all,!spam,!virus"

Syntax: user_access_default string

user_alias - Number of aliases accounts can create

This setting specifies the maximum number of account aliases an account in this domain is allowed to create. The format of these aliases is specified in the file specified by the [g_user_alias_file](#) setting.

Default is disabled (0) which disables the alias creation interface in the user.cgi pages.

Syntax: user_alias int

user_auto - Auto create users when a login attempt occurs

Exceedingly dangerous setting only intended for closed systems not open to internet! Accounts are created when someone tries to login via imap/pop etc... If the account already exists it is not modified unless it is currently set with the password defined in user_auto_pass (broadsoft feature)

Syntax: user_auto bool

user_auto_pass - Auto create users with this password on message delivery

Exceedingly dangerous setting only intended for closed systems not open to internet! Accounts are created on message delivery if they don't exist with the specified password. Broadsoft Extentsion account autoprovisioning (Broadsoft feature)

Syntax: user_auto_pass string

user_centipaid - see [CentiPaid.htm](#)

Specifies the various CentiPaid options a user is allowed to configure for themselves.

Syntax: user_centipaid string

user_list_quota - Number of mailing lists users can create

This setting configures the number of mailing lists a user of this domain can create. See also g_user_list_quota which can set quota globally or by user group. Also the list_quota authent field can set quota per user.

Syntax: user_list_quota int

user_max - Maximum user allowed in this domain

This setting specifies the maximum number of users in this domain. Domain admins and users are blocked from adding more users than specified in this setting. The admin can still add users.

Syntax: user_max int

user_send_max - Maximum number of emails per day (requires SMTP AUTH)

This setting specifies the maximum number of messages an account in this domain is allowed to send in a day.

Syntax: user_send_max int

user_sms - Enable users to setup SMS notifications

This setting allows users to setup an SMS notification of email whose subject matches the user defined keyword.

Syntax: user_sms bool

user_sms_quota - Number of sms messages per account

This setting allows you to configure either a quota or a 'number of credits' system for SMS messages. This setting has 2 parameters 'initial' and 'period'.

If you set 'period' to 0 then 'initial' is the number of credits a user will begin with they are decremented when they send an SMS and not increased unless you increase them manually.

If you set 'period' to any value other than 0, then 'initial' is the users quota, which is re-set after the time period specified by 'period'. Valid 'period' values are a number of hours, or suffix the value with d, w or y to indicate a number of days, weeks or years respectively, eg: 5w equals 5 weeks.

Syntax: user_sms_quota initial=int period=string

user_status_send - How often to send user status messages (0 = never)

When the user enables friends then this setting will send them a regular report on what is pending and what filter rules have done.

Syntax: user_status_send int

See also: [friends_at_rcpt](#), [friends_pending_name](#), [friends_url](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_bounce_friend](#), [g_friends_daemon_ok](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_lang_auto](#), [g_friends_pending_keep](#), [g_friends_pending_max](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_safer](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_global_add](#), [g_friends_global_exclude](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_long](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_byemail](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_friends_debug1](#), [g_imap_friends](#), [g_quota_friends](#), [g_spam_probe_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

web_access_ip - Restrict access to web ports based on ip

Specifies a list of ports and a wildcard list of valid ip addresses who can connect to those ports.

Syntax: web_access_ip ports=string ip=string

web_path - Path to web admin pages

Path to web admin and user self management pages. This setting allows you to give each domain a different set of pages and thus a different look and feel. To enable this, copy the entire 'web' directory then set web_path to the path of the copied files. Lastly modify the copied files to have the new look and feel you desire.

Syntax: web_path string

web_url_path - Url to path translation with access specifier

This lets you set up aliases and translations of urls partly based on the access rights of the user.

Syntax: web_url_path url=string path=string access=string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_known_skip](#), [g_web_php_exe](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#), [g_web_utf8](#)

webdav_quota - Webdav quota per user in this domain, e.g. 100mb

Limits the webdav storage area for this user, default is the users email quota

Syntax: webdav_quota string

webmail_host - The IP address of the mail server

SurgeMail sets the imaphost/smtphost address in surgehost.ini for WebMail. First it checks for a virtual domain ip, then it checks for a specifically bound ip address in the imap/smtp port settings, if neither are specified it defaults to 127.0.0.1. The webmail_host will override this process and assign a value directly. You require this when using a Smart Router or Load Balancer, please read [this](#).

Syntax: webmail_host string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_known_skip](#), [g_web_php_exe](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#),

[g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#), [g_web_utf8](#)

webmail_url - Url to the WebMail cgi

If WebMail is not in the default place and/or is not on the SurgeMail machine then this setting tells SurgeMail where it is so links to WebMail from SurgeMail function correctly.

Syntax: webmail_url string

See also: [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_known_skip](#), [g_web_php_exe](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#), [g_web_utf8](#)

webmail_urladd - Url data to append to WebMail auto-login link

This setting allows you to specify additional information and settings which are passed to WebMail when SurgeMail links to it. Example: Different colors to use for this domain.

Syntax: webmail_urladd string

See also: [webmail_url](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_known_skip](#), [g_web_php_exe](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#), [g_web_utf8](#)

webmail_workarea - Path to WebMail workarea

If WebMail is not installed in the default location on this SurgeMail machine this setting tells SurgeMail where to find it.

Syntax: webmail_workarea string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_known_skip](#), [g_web_php_exe](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#), [g_web_utf8](#)

xfile_url - Url to xfile files (see surgeplus utility)

Use to override the url that users are told they can access their shared SurgePlus files at via a web browser. The default location is on the port specified by the webmail_port setting.

Syntax: xfile_url string

Example: https://your.domain.name:7443

See also: [old_xfile](#), [g_webmail_port](#), [g_xfile_allow](#), [g_surgeplus_web_port](#), [g_surgeplus_web_url](#)

Global Settings

Note: Most 'matching' settings take **wild card lists** as parameters, for example "fred*" will match "freddy" and "Fred@bob". And "1.2.*.2.3.*" will match 1.2.4.4 and 2.3.99.100. Many settings will also accept a ! as a "not", and are processed from left to right. eg "!*,127.*,10.*" would first "deny all" then try and match on any 127.* or 10.* domains. Settings using ip's will take ranges also like 10.0.1-120.5 and also support CIDR notation eg 10.10.1.32/27.

You can read about CIDR notation here http://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing and there is an online CIDR calculator here <http://www.subnet-calculator.com/cidr.php>.

g_access_group - Access groups

Access rules defining groups of IP addresses with certain POP, IMAP and SMTP privileges. When a user is authenticated access is checked against group membership defined in the "mailaccess" field in the authentication database. See [accounts](#) for more information.

eg. this could allow you to charge webmail users for pop access privileges:

```
g_access_group group=paid_user access_pop=* access_imap=* access_smtp=*
g_access_group group=free_user access_pop=webmail.svr.ip access_imap=webmail.svr.ip
access_smtp=webmail.svr.ip
```

with "Access type" set to "free_user" on accounts page or equivalently in nauth authentication database:
marijn@mydomain.com: { ssh } tVANQo...: created="1060034937" mailaccess="free_user" ...

To prevent webmail access for some users you would do this:

```
g_access_group_default "normal"
g_access_group group="normal" access_pop="" access_imap="" access_smtp=""
g_access_group group="nowebmail" access_pop="*,!webmail.ip" access_imap="*,!webmail.ip"
access_smtp=""
```

And put the users you want to limit in a group called 'nowebmail' e.g.

```
lookup fred@domain
+OK fred@domain config 0 mailaccess="nowebmail"
```

Syntax: g_access_group group=string access_pop=string access_imap=string access_smtp=string
access_incoming=string

See also: [g_user_alias](#), [g_user_blogs](#), [g_user_access](#), [g_user_sms_quota](#), [g_user_send_max](#), [g_user_list_quota](#)

g_access_group_default - Access group defaults

Access group defaults for users with no access groups set. (must be used in conjunction with g_access_group)

Syntax: g_access_group_default string

g_acctlog_authonly - Log sending usage based on authenticated accounts only and ignore "MAIL FROM" address - which may be fake!!

This setting has no further documentation currently available

Syntax: g_acctlog_authonly bool

g_acctlog_noaliases - Don't log redirection & aliases as sending mail as a result of redirection / forwarding (means you will not log account forwarding usage)

This setting has no further documentation currently available

Syntax: g_acctlog_noaliases bool

g_acctlog_sum_inactive - Summarise local accounts that have not logged in yet as not_loggedin_yet@domain.com

This setting has no further documentation currently available

Syntax: g_acctlog_sum_inactive bool

g_admin_access - Allow / Restrict domain admin access to features based on [g_access_group](#)

g_admin_access group="wildcard" access="list"

This setting matches the g_access_group the admin is in to the wildcard specified and applies the specified access list to that domain admin, giving / restricting thier access to certain features. The list may include any of the following:

Value	Result
alias	Access to domain users "Alias" page and features.
asbam	Access to the "ASbam" page and features.
blog	Access to the "Blogs" page and features.
bulletins	Access to the "Bulletins" page and features.
centipaid	Access to domain users "Centipaid" page and features.
enotify	Access to domain users "Email Notification" page and features.
exceptions	Access to domain users "Exceptions" page.
friends	Access to domain users "Friends" pages, and system.
fwd	Access to domain users "Forwarding" features, forwarding, auto-responder.
fwdonly	Access to domain users "Forwarding" features, forwarding
lists	Access to the "Lists" page and features.
log	Access to domain users "Log" page.
mailbox	Access to domain users "Mailbox" page, view mailbox, setup rules.
sms	Access to domain users "Sms" page.
spam	Access to domain users "Spam" page, and SmiteSpam and Asbam processing of messages.
spampriv	Access to domain users "Spam" pages' spam private feature
spf	Access to domain users "Spf" page and features.
usage	Access to the "Usage" button, which shows a domain users usage.
users	Access to the "Users" page and features.
redirect	Access to the "Redirect" page and settings.
redirect_cc	Access to the "Redirect CC" page and settings.

In addition you can prefix any of the above with ! to deny access. There are two other special case values, "all" and "none" which mean exactly what they say, access to "all" or "none" of the features.

Example:

g_admin_access group="simple" access="all,!users,!reports"

The above setting gives admins in the 'simple' group access to all the features except the users and reports features.

Syntax: g_admin_access group=string access=string

g_admin_access_default - Default features granted to domain admins

This setting is a default access list for all domain admins on the server, it is specified in the same maner as the [g_admin_access](#) settings 'access' parameter. eg:

`g_user_access_default` "all,!users,!reports"

Syntax: `g_admin_access_default` string

g_admin_guesses - Number of guesses allowed for admin.

Syntax: `g_admin_guesses` "number"

This sets the number of guesses allowed for the admin username/password. Once this has been reached the ip is banned.

Syntax: `g_admin_guesses` int

See also: [g_admin_ip](#), [g_admin_localhost](#), [g_admin_access](#), [g_admin_access_default](#), [g_admin_utoke n_expire](#), [g_admin_utoke n_idle](#)

g_admin_ip - Admin IP access

Mask of valid IP addresses for admin users (default *), this is a security setting you can use to restrict remote web admin access to trusted IP addresses. One is always allowed to use manage SurgeMail using 127.0.0.1 regardless of whether this is explicitly specified.

eg. To restrict to local network as per net mask

`g_admin_ip` "10.0.0.*,10.1.2.*"

Syntax: `g_admin_ip` string

g_admin_localhost - Allow localhost web admin without user/pass

Allows a localhost connection to access the web admin port without using the administrator username / password. This is good if you keep forgetting the admin password like I do.

Syntax: `g_admin_localhost` bool

See also: [g_admin_ip](#), [g_admin_guesses](#), [g_admin_access](#), [g_admin_access_default](#), [g_admin_utoke n_expire](#), [g_admin_utoke n_idle](#)

g_admin_utoke n_expire - Length of time a web admin session is valid for

This setting has no further documentation currently available

Syntax: `g_admin_utoke n_expire` int

g_admin_utoke n_idle - Length of time a web admin session may remain idle for

This setting has no further documentation currently available

Syntax: `g_admin_utoke n_idle` int

g_alias_login_disable - Disable user login as alias

Stops the user login to pop or imap as the alias account

Syntax: `g_alias_login_disable` bool

g_allow_bodyless - Allow bodyless email

This will allow bodyless email to be accepted. These are usually spam. In particular Norton Antivirus in autoprotect mode closes the POP link which makes it appear that SurgeMail has terminated the connection when a bodyless email is encountered.

Syntax: `g_allow_bodyless` bool

g_allow_passzip_from - A list of addresses to allow unmonitorable archive messages to be sent from
These may of course contain viruses as they cannot be scanned, but some people still need to be able to accept such files.

Syntax: `g_allow_passzip_from` string

g_allow_passzip_to - A list of addresses to allow unmonitorable archive messages to be sent to. These may of course contain viruses as they cannot be scanned, but some people still need to be able to accept such files.

Syntax: g_allow_passzip_to string

g_allow_user_authent_field_get - A space separated list of authent process fields that users are allowed to view for themselves using the POP xauthent_field_get command

This provides limited access to the user database for applications like webmail and surgeplus.

Syntax: g_allow_user_authent_field_get string

See also: [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#)

g_allow_user_authent_field_set - A space separated list of authent process fields that users are allowed to set for themselves using the POP xauthent_field_set command

This provides limited access to the user database for applications like webmail and surgeplus.

Syntax: g_allow_user_authent_field_set string

See also: [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#)

g_archive - Archive delivered mail

Archive rules allowing all mail delivered to be archived to either:

- **Fixed size rotating archive** - use this if you want to be able to get back a particular message that has recently passed through the server but you do not want the mail archives to be able to grow too large
- **History archive** of a fixed (or unlimited) duration that can grow as much as the disk space available. Use this if you need to archive say all mail sent to / from a particular customer for the last year.

The archive is stored as a directory containing bucket files. This allows you to retrieve messages that have been delivered if you need to retrieve a particular message for any reason. To retrieve a message this needs to be extracted manually from the archive files manually using a text editor or your own script. The maximum bucket size (default if 1Mb) of the archive and the maximum individual message size can be set.

Filtering is done based upon wildcard destination and source addresses and subject. These fields provide a logical AND, with a blank field matching the default "". A specific email may match multiple archive rules, and will be archived in each archive in this case. Also note that if a match is part of a larger string the match string should have wildcards surrounding it. eg: to match "important business" in the subject "Very important business for you" you should specify "*important business*".

eg. To catch all email delivered from domain.com you would specify:

g_archive to="" from="*@domain.com" subject="" path="c:\mailarchive" size="10mb" maxitem="10k"

You can also select whether the archiving rule is triggered before or after any filtering that is applied such as virus or spam filtering using the early flag. This can be useful to capture the original source of viruses or spam for testing purposes.

Syntax: g_archive to=string from=string path=string subject=string size=string maxitem=string keep=string early=bool owner=string

g_archive_bucketsize - Size for archive bucket files. Default is 1mb

Sets the size of the archive buckets used by the circular archives. If set too large then editing the buckets manually is awkward.

Syntax: g_archive_bucketsize int

See also: [g_archive](#), [g_archive_tcpip](#), [g_archive_tcpip_host](#), [g_archive_early](#), [g_archive_on_delete](#), [g_archive_on_delete_dir](#), [g_archive_files](#)

g_archive_early - Apply all archive rules before content filtering is applied (obsolete)

This will apply the archive rules before content filtering is applied. This can be user to capture the source message if it is getting stored or bounced unnecessarily by any of the SurgeMail filters. The early flag on individual archive rules should be used instead of this setting.

Syntax: g_archive_early bool

g_archive_files - Archive attachments to a directory

Each message to the named account will have it's attachments removed and placed in the named directory. The path can contain the symbols \$month\$ \$year\$ \$day\$ \$second\$. The 'second' is only within this day. Together these variables can be used to ensure a unique path is used for each file if the names might conflict. Use g_redirect_cc to archive email going to an existing account because if you set 'to' equal to a real account then the real account will stop receiving messages!

Syntax: g_archive_files path=string to=string files=string

g_archive_on_delete - Don't delete user files, archive them to g_archive_on_delete_dir
When deleting a user, archive the users files in the g_archive_on_delete_dir archive directory.

Syntax: g_archive_on_delete bool

g_archive_on_delete_dir - Directory to archive user files to on delete
Directory to archive deleted users files to. Defaults to 'archive' in the SurgeMail installation folder.

Syntax: g_archive_on_delete_dir string

g_archive_tcpip - Rules for TCPIP archive process
Contact netwin for more details of this mechanism if you wish to use it.

Syntax: g_archive_tcpip to=string from=string path=string dom=string

g_archive_tcpip_host - Host to send archive data too
When using an archive server this defines the host that is running the archive server. Contact netwin if you need more info on this feature.

Syntax: g_archive_tcpip_host string

g_aspam_headers - Add aspam information messages to messages.
Adds informational aspam headers to all messages.

Syntax: g_aspam_headers bool

See also: [g_aspam_need_ip](#)

g_aspam_need_ip - Require good matches to match external ip address
This prevents poluted bad messages in aspam_good causing spam to bypass the filters, but reduces effectiveness of the notspam address.

Syntax: g_aspam_need_ip bool

See also: [g_aspam_headers](#)

g_assume_created_epoch - If user has no 'created' field assume they were created an arbitrarily large time in the past
This setting effect the g_disable_smtp_after and g_delete_user_after settings which, by default, ignore users who have not logged in and have no created field.

Syntax: g_assume_created_epoch bool

g_atrn_client - Define a rule for fetching email

This is the setting for clients to define to fetch mail from an upstream server. Typically this is done on the special port 366, to specify another port use host:port in the host setting. E.g.

host="smtp.upstream.com:25"

Syntax: g_atrn_client domain=string user=string pass=string host=string

See also: [g_atrn_server](#), [g_atrn_port](#)

g_atrn_port - Port to listen for 'atrn' (On Demand Relay) requests

See g_atrn_server for more details, the default is port 366, atrn is not obeyed on port 25

Syntax: g_atrn_port string

See also: [g_atrn_server](#), [g_atrn_client](#)

g_atrn_server - On Demand Mail Relay settings to define user/pass for clients to fetch mail

This allows a client on a dynamic IP to connect and request mail for a specific domain after authenticating by using the ATRN command. Typically this is done on the special port 366

Syntax: g_atrn_server domain=string user=string pass=string

See also: [g_atrn_client](#), [g_atrn_port](#)

g_auth_hide - Disable SMTP Authentication

Per default SMTP authentication is enabled. If a user matches this IP range/list they will NOT be shown the ESMTP extension for SMTP authentication. This will usually stop the mail client from prompting the user for authentication. We STRONGLY recommend you do NOT use this feature. It is much better to let users authenticate when sending email.

Syntax: g_auth_hide string

g_auth_norelay - Ignore SMTP auth for relaying purposes

This means relaying only occurs if g_relay_allow_ip matches

Syntax: g_auth_norelay bool

g_auth_skipgateway - Skip gateway rules if we get a proxy SMTP auth command

Skip gateway rules if we get a proxy SMTP auth command. This is not for general use. It can be used if you are using SurgeMail in front of another mail server with a wild card gateway to gateway all domains to a back end mail server. Then an authenticated user is a local user trying to send out so the gateway rules are ignored. (this is strongly not recommended)

Syntax: g_auth_skipgateway bool

g_authent_allow_badascii - Allow ascii chars outside the range 32 < 127

By default ascii characters < 32 and >= 127 are blocked as invalid. If you require these characters set this to TRUE.

Syntax: g_authent_allow_badascii bool

g_authent_always - Always lookup user, so virtual domains can exist just in authent module

Always lookup user, so virtual domains can exist just in authent module. This allows you to support 10,000 domains on one system without a 'huge' ini file. Be careful to not create/remove real domains with the same name as existing domains that only exist in the authent database as the 'drop files/inboxes' will move when this occurs and existing mail will vanish.

Syntax: `g_authent_always` bool

g_authent_any - Restore buggy behaviour of looking up users in domains that don't exist

Previously surgemail would lookup a user even if the domain in question did not exist, if you need to restore this odd behaviour then you can use this setting...

Syntax: `g_authent_any` bool

g_authent_cachebad - Cache life of failed authent lookups

Set the life in seconds that the cached failed lookups can be used, default 60 seconds. Best left alone unless your server is being hit by thousands of failed lookups and your authent module is slow.

Syntax: `g_authent_cachebad` int

g_authent_cachelife - Cache life of successful authent lookups

Set the life in seconds that successful cached lookups can be used, default 2 hours. Best left alone.

Syntax: `g_authent_cachelife` int

g_authent_cachesize - Size of the authent cache

Set the size of the authent cache, default is 500 entries. Generally best left alone.

Syntax: `g_authent_cachesize` int

g_authent_domain - Authent domain

If this is 'true', the virtual domain name is appended to the username before it is passed to the authent process. This lets the authent process deal with virtual domains. As a general rule, this should ALWAYS be true.

Syntax: `g_authent_domain` bool

g_authent_encrypt_key - Encryption key for ccnumber auth field

Not for general use currently, used to partially obscure credit card info when stored in the authent module.

Syntax: `g_authent_encrypt_key` string

g_authent_info - Authent info

Defines a piece of information to store about the user in the user database (phone number, name, address etc). Each piece of information is given a name, a field, an access mode, a default and a type. The name defines what appears in the web management display. The field is what is sent to the `authent_process`. The access mode can be one of the following: `user`, `domadmin`, or `admin`, `createonly`, `none`. The default is what value is assigned upon creation of a new user. The type can be one of: `date`, `readonly`, `encrypt` or any custom string which you want to check for or match on the `na_details.htm` page with a template function like: `||ifequal||user_info_type||custom||` .. do things .. `||endif||`

An access mode of 'admin' means that only the system admin can see the information, 'domadmin' means the sysadmin and any domain admin can see the information, 'user' means the user can see the information, 'createonly' means the user sets the information at creation time but cannot see it after that and 'none' ensures that no-one can see or modify the information (used for information that is handled by SurgeMail itself, either through the interface or otherwise)

e.g.
`g_authent_info name="Phone Number" field="phone" access="user" default="" type=""`

See [here](#) for a complete list of default settings.

Syntax: `g_authent_info` name=string field=string access=string default=string type=string

g_authent_info_grp - Fields to show to users in this group

Specifies the authent fields this user group is allowed to see and change. This applies only to the fields visible on the account properties page and the domain admin "Users" page it cannot be used to prevent

access to fields which are managed by the web interface i.e. 'fwd'

Syntax: g_authent_info_grp group=string fields=string tag=string

g_authent_ip - Authent Lookup IP numbers via authent modules - enables relaying

If enabled each connecting IP address will be looked up in your user database as x.x.x.x@ip eg: "127.0.0.1@ip" and if the user is found then relaying is allowed and if 'send_limit="nn"' is defined then that will set the tarpit send limit for that user.

For per IP tarpit limits to work you need to define the g_tarpit_max and g_tarpit_max_remote settings. And g_tarpit_drop to make the limit effective.

Syntax: g_authent_ip bool

g_authent_last_login - Store users last login time in the database

This setting will cause the authent field 'last_login' to be updated when a user logs in. The field is set to a timestamp which is 'the number of seconds since midnight January 1, 1970'. This field is updated 'at most' once every 24 hours. Other features i.e. delete_user_after and disable_smtp_after will look for this field.

Syntax: g_authent_last_login bool

g_authent_logall - Turns on logging of authent requests

If enabled, authentication requests are logged in mail.log as "<day> <time> Authent[<action> <info>]".

Syntax: g_authent_logall bool

g_authent_number - Authent number

The number of concurrent authent processes to run. If you are using a slow external authent module (e.g. sql) then it is probably worth running 3-4, there is no need to have more than 1 when using nauth.exe. (Default = 1)

Syntax: g_authent_number int

g_authent_process - Authent process

The command line of a NetWin authentication module. You can use one of our standard modules for LDAP, ODBCAuth, MySQL etc or write your own. For more information on these modules see the authentication section of the [manual](#) .

This will typically be something like:

g_authent_process "E:\surgemail\nwauth.exe -path E:\surgemail"

or

g_authent_process "/usr/local/surgemail/nwauth -path /usr/local/surgemail"

Syntax: g_authent_process string

g_authent_restart - Cycle auth modules every 1000 lookups

This is useful if there are resource allocation issues in the authentication module. Eg ODBCAuth

Syntax: g_authent_restart bool

g_authent_single - Allow local users with a single quote char in their name

This let's users exist who contain the single quote ' character. It is not supported with some authent modules though, nauth does allow it.

Syntax: g_authent_single bool

g_authent_spaces - Allow spaces in passwords DO NOT USE

Not supported for most authent modules, requires nauth 4.0r or later, If you have already got users with spaces in their passwords and you turn this setting on, they will no longer be able to login until they reset their passwords. Authent module must support slash encoding, for nauth add -spaces to command line

Syntax: g_authent_spaces bool

g_authent_strip_domain - Strip domain for authent lookups

Use when your database expects one 'primary' domain to do lookups without a domain name then SurgeMail will strip that domain only from lookups. Typically this is only necessary with old DMail authent modules.

Syntax: g_authent_strip_domain string

g_authent_timeout - Timeout for authent response

Timeout for authent response, default 60 seconds.

Syntax: g_authent_timeout int

g_autologin_file - File to use to share auto login information on NFS based cluster

This allows webmail to autologin when using an nfs based cluster and a load sharing device.

Syntax: g_autologin_file string

g_autologin_imap_disable - Disable IMAP based autologins

IMAP autologins allow autologin to surgeweb.

Syntax: g_autologin_imap_disable bool

g_autologin_pop - Enables WebMail Autologin using POP when on another server

Webmail needs the ability to automatically login to SurgeMail to changes passwords etc. This setting will do this via an extension to the pop protocol allowing WebMail to autologin whilst running on another server. (Normally this is done using a temporary file)

Syntax: g_autologin_pop bool

g_bad_login_allow - Number of consecutive bad logins for a user before blocking that user

Number of consecutive bad logins for a user before blocking that user.

Syntax: g_bad_login_allow int

g_bad_login_ip_allow - Number of bad logins from an IP before blocking that IP

Number of bad logins from a single IP before blocking that IP.

Syntax: g_bad_login_ip_allow int

g_bad_login_ip_ignore - IP address(es) to ignore bad logins from

Use for webmail system or other local gateway to stop bad login counter from locking out all users.

Syntax: g_bad_login_ip_ignore string

See also: [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_badfrom_noip](#), [g_badfrom_noip_temp](#), [g_badfrom_check](#), [g_badfrom_stamp](#), [g_badfrom_badmx](#), [g_badfrom_from](#), [g_badfrom_whitelist](#)

g_bad_login_mins - Minutes to block login for, if consecutive bad ones received

Minutes to block login for, if consecutive g_badlogin_allow or g_badlogin_ip_allow bad logins received=.

Syntax: g_bad_login_mins int

g_badfrom_badmx - Drop message if this MX

If mx host is one of these addresses then drop the message, it's definitely spam (e.g. 127.*).

Syntax: g_badfrom_badmx string

g_badfrom_check - Check if 'from' envelope can be delivered to

If this is set to "true" then SurgeMail will connect back to the envelope 'from' address and check that the address is valid, a cache is used to improve performance, if it cannot connect then the message is bounced as probable spam. It's nicer to use the following setting "g_badfrom_stamp" as well, then if SurgeMail cannot connect back or the user is invalid then a header is added to indicate this, and our SmiteSpam rules will use this to increase the spam weighting.

You can use g_spam_allow to exempt an IP from this check as well as g_badfrom_whitelist for a domain. Please note that by default SurgeMail uses a blank mail from to do its check.

MAIL FROM: <>

Some servers might reject this, though they shouldn't because it's a standard bounce, however if they do you can use [g_badfrom_from](#) to set a mail from address to be used for this check.

Syntax: g_badfrom_check bool

g_badfrom_from - Mail from account for g_badfrom_check

From to use when doing the g_badfrom_check check, not normally needed, if set must be set to valid account.

Syntax: g_badfrom_from string

g_badfrom_noip - Check envelope from domain exists and is a valid IP number

Check envelope from domain exists and is a valid ip number, if not bounce message.

Syntax: g_badfrom_noip bool

g_badfrom_noip_temp - Makes g_badfrom_noip return a temporary error instead of a 501 error
Use g_verify_mx_skip to bypass/whitelist ip addresses from this check

Syntax: g_badfrom_noip_temp bool

g_badfrom_stamp - If 'g_badfrom_check' is bad then stamp a header on the message

g_badfrom_check must also be set to true. If this is set to "true" then SurgeMail will connect back to the envelope 'from' address and check that the address is valid, a cache is used to improve performance, if it cannot connect then a header is added to indicate this, and our SmiteSpam rules will use this to increase the spam weighting.

Syntax: g_badfrom_stamp bool

g_badfrom_whitelist - Whitelist of domains to skip from checks

Whitelist of "from" address domains to skip g_badfrom_* checks.

eg.

g_badfrom_whitelist "specialdomain.com"

Syntax: g_badfrom_whitelist string

g_ban_blackhole - Leave connected but reject all recipients without looking them up

Leave connected but reject all recipients without looking them up. This is good of dealing with high volume spammers without wasting resources doing user lookups.

Syntax: g_ban_blackhole bool

g_ban_from - Ban any matching MAIL FROM: envelope

Same as 'ban_helo' but applies to the from (return address) part of the mail envelope. This is NOT the same as the from/sender header in the message itself!!! This equates to the 'Return-path:' header that the mail server adds.

Syntax: g_ban_from string

g_ban_helo - Ban any machine that gives a matching 'helo' string

This is a simple spam protection system to block known spam/problem users based on the 'helo' name they send to your system. This name is recorded in the 'received' header along with the IP address. This name is very easy to 'fake' so is not a high security level of protection, but it is simple for stopping stupid robots etc, that have gone insane.

Example: *junkmail.com

Syntax: g_ban_helo string

g_ban_rcpt - Ban any matching RCPT TO: envelope

Same as 'ban_helo' but applies to the recipient part of the envelope (destination users) this is NOT the same as the 'To:' header in the message itself!!! This can sometimes be used to block really simple spamming programs that always send to the same invalid users.

Syntax: g_ban_rcpt string

g_bank_debug - Log request to bank server

Use when trying to debug the g_bank_url post/response

Syntax: g_bank_debug bool

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_group](#)

g_bank_group - Create price groups with descriptions

See g_bank_url for details

Syntax: g_bank_group group=string price=string desc=string

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_debug](#)

g_bank_log - Log lines matching this in response.

See g_bank_url for details

Syntax: g_bank_log string

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_debug](#), [g_bank_group](#)

g_bank_ok - Find this in response, if found then charge was successful

See g_bank_url for details

Syntax: g_bank_ok string

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_pass](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_debug](#), [g_bank_group](#)

g_bank_pass - Password for authenticated web request to banks system

See g_bank_url for details

Syntax: g_bank_pass string

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_debug](#), [g_bank_group](#)

g_bank_reason - This line is returned to user if it is found

See g_bank_url for details

Syntax: g_bank_reason string

See also: [g_bank_url](#), [g_bank_user](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_log](#), [g_bank_debug](#), [g_bank_group](#)

g_bank_url - URL to charge a credit card (experimental)

This allows automated monthly charging of users

Syntax: g_bank_url string

See also: [g_bank_user](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_debug](#), [g_bank_group](#)

g_bank_user - Username for authenticated web request to banks system

See [g_bank_url](#) for details

Syntax: **g_bank_user** string

See also: [g_bank_url](#), [g_bank_pass](#), [g_bank_ok](#), [g_bank_reason](#), [g_bank_log](#), [g_bank_debug](#), [g_bank_group](#)

g_bind_byfromip - Bind outgoing SMTP connections to the specified IP based on the sender IP

This setting has no further documentation currently available

Syntax: **g_bind_byfromip** fromip=string bindip=string

g_bind_from - Bind outgoing SMTP connections based on 'from' envelope

Bind outgoing SMTP connections based on the IP of the virtual domain in 'from' envelope. This is only useful if you are using IP based virtual domains.

Syntax: **g_bind_from** bool

g_bind_incoming - Bind outgoing SMTP connections based on incoming ip address

So if the incoming mail came in on interface address 1.2.3.4 then that same address is used to send the email

Syntax: **g_bind_incoming** bool

g_bind_out - Bind outgoing smtp connections to IP

Bind outgoing smtp connections to this IP number.

Syntax: **g_bind_out** string

g_black_above - Level for spam detection for **g_black_count**

Level for spam detection for blacklisting IP number e.g. 7.

Syntax: **g_black_above** int

g_black_count - Blacklist sender IP based on spam sent

Number of spam in a row before IP blacklisted for 30 minutes eg: 30 (default = disabled)

Syntax: **g_black_count** int

g_black_isspam - Blacklist ip address for any spam training event

This setting has no further documentation currently available

Syntax: **g_black_isspam** bool

g_black_to - Blacklist sender IP based on catch addresses

Blacklist senders IP address for 30 minutes if they deliver to these spam catch email addresses.

eg. **g_black_to** "smith@mydomain.com,catcher@myotherdomain.com"

Syntax: **g_black_to** string

g_black_white - Whitelist to prevent blacklisting, e.g. 1.2.3.*,mail*.aol.com

This setting has no further documentation currently available

Syntax: **g_black_white** string

g_block_files - Block certain attachments

Allow you to block any mail with certain files attached.

g_block_files "*"*.exe,**.cmd,**.com"

Syntax: g_block_files string

See also: [g_block_wild](#), [g_block_skip](#), [g_block_longok](#), [g_debug_block](#)

g_block_longok - If true allow long file names (more than 180 char)

By default files names over this length are ALWAYS blocked if g_block_files is used, in rare situations these are not just viruses attempting to get around the filter.

Syntax: g_block_longok bool

See also: [g_block_wild](#), [g_block_files](#), [g_block_skip](#), [g_debug_block](#)

g_block_skip - From or To address to bypass g_block_files

Some users will need to send various attachments, these users are exempt to the g_block_files rule

Syntax: g_block_skip string

See also: [g_block_wild](#), [g_block_files](#), [g_block_longok](#), [g_debug_block](#)

g_block_wild - Block wildcards in usernames

Block the '*' wildcard character in usernames.

Syntax: g_block_wild bool

g_blogs_allow_links - Allow users to post comments that contain urls

Due to widespread abuse of blogs this is not recommended.

Syntax: g_blogs_allow_links bool

g_blogs_cleanup_links - Delete existing posts that contain urls

This setting will help cleanup existing spam postings to your users blogs.

Syntax: g_blogs_cleanup_links bool

g_blogs_comment_rev - Show blog comments newest first

Helps if there are lots of comments, this is a global setting not per blog..

Syntax: g_blogs_comment_rev bool

g_blogs_default_template - Default template set that is used by newly created blogs

This setting can have a value of the name of any directory in the SurgeMail blogtpl directory

Syntax: g_blogs_default_template string

g_blogs_donly - Only list blogs in a users domain

By default all blogs in all domains are listed/shown to the user. This setting causes it to only list blogs in the users domain.

Syntax: g_blogs_donly bool

g_blogs_enable - Surgemail blogs

Allow users to create blogs

Syntax: g_blogs_enable bool

g_blogs_image_optional - Allow users to specify if image verification is required for comments
By default image verification is now required, this prevents spammers from abusing the many 'test' blogs set up by your users.

Syntax: **g_blogs_image_optional** bool

g_blogs_max_per_user - Maximum number of blogs per user
Maximum number of blogs per user, default is 5

Syntax: **g_blogs_max_per_user** int

See also: [blogs_max_per_user](#), [g_user_blogs](#)

g_blogs_maximum_image_size - Default maximum image size
Images larger than this (in largest dimension) that are posted to blogs are scaled down, default is 390, per blog setting can override this.

Syntax: **g_blogs_maximum_image_size** int

g_blogs_maximum_image_width - Default maximum image width
Images larger than this that are posted to blogs are scaled down, default is 390, per blog setting can override this.

Syntax: **g_blogs_maximum_image_width** int

g_blogs_maximum_items_in_top_page - Maximum number of items on the top blog page
Maximum number of post bodies to appear on a blog top page, default is 10

Syntax: **g_blogs_maximum_items_in_top_page** int

g_blogs_no_suffix - Shortens URL, **url_blogs** must be defined for each domain
This shortens <http://a.com/blog/juggling> to <http://a.com/juggling>, but does require that you define a specific name for the blogs in the domain based **url_blogs** setting

Syntax: **g_blogs_no_suffix** bool

g_blogs_not_global - Only allows access to a blog on the domain it is defined on
Only allows access to a blog on the domain it is defined on, this is not recommended. (probably want to use **g_blogs_not_unique**, **g_blogs_domonly** too)

Syntax: **g_blogs_not_global** bool

g_blogs_not_unique - Allow the same blog name in multiple domains
If set you can create different blogs with the same name in different virtual domains, this is not recommended.

Syntax: **g_blogs_not_unique** bool

g_blogs_ping - Sites to ping on each post
Host and path to ping on each blog post. eg: **host=rpc.weblog.com path=/RPC2**

Syntax: **g_blogs_ping** host=string path=string

g_blogs_sub_domain_prefix - Prefix to use instead of blogs. for blog subdomains. use ! to have no prefix.
Experimental feature do not use

Syntax: **g_blogs_sub_domain_prefix** string

g_blogs_use_sub_domains - Make blogs accessible at http://blog_name.domain/

If you're DNS entry supports it, turn on this setting to make blogs accessible at `http://blog_name.blogs.domain/` instead of `http://domain/blogs/blog_name`

Syntax: `g_blogs_use_sub_domains` bool

g_body_filter - Enable user email body filtering

Allows the user to configure filters which filter the body of incoming messages

Syntax: `g_body_filter` bool

g_bomb_max - Max messages to a single address per hour

Simple system to prevent intentional or more likely, accidental mail loops or mail bombs where thousands of Emails are sent to a single user. A setting in the range of 100-1000 is generally good depending on your sensitivity to incorrectly blocking real mail. We suggest 1000 is a good setting if you are unsure.

This counts the messages from a single IP address to a single recipient. If a single IP sends more than this many messages to any single recipient then they will be tarpitted (slowed down and rejected).

Use `spam_allow ip.address.list` to over-ride the limit for known local systems that might exceed this limit (unlikely anything will).

Syntax: `g_bomb_max` int

g_bomb_max_from - Max msgs from a single email address/hour

Max msgs from a single email address/hour.

Syntax: `g_bomb_max_from` int

g_bounce_bind - Use a specific ip address for outgoing bounces

Some RBL sites blacklist machines for sending bounces, which is probably a good thing. But even with spf running your server may occasionally send a bounce to a forged address, and so you can use an alternate ip address for these bounces to avoid blacklisting your main mail server address. First you must assign the ip address to your network interface etc

Syntax: `g_bounce_bind` string

g_bounce_disable - Bounce Disable

Disable all bounces. This is particularly useful when under spam attack. This is for outgoing bounces it stops SurgeMail generating bounces it won't affect incoming bounces from other servers.

example:

`g_bounce_disable "true"`

Syntax: `g_bounce_disable` bool

g_bounce_limit - Max size of bounce messages

Max size in bytes of message to send back as bounce message is truncated if necessary.

Syntax: `g_bounce_limit` int

g_bounce_nodrop - Enables locally generated bounces for non local users

This setting makes bounces occur normally, the reason bounces are normally dropped for non local users is that they are almost always spam bouncing off another server due to forwarding settings, and as such sending a bounce email will get your server black listed, so we decided it was best to drop them by default since they are rarely useful. Turn this setting on at your own risk :-). Instead use `g_bounce_to` to list domains that it is safe to bounce to.

Syntax: `g_bounce_nodrop` bool

g_bounce_redirect - Send all bounces to a local address

This can be used to avoid 'back scatter' which can get your server listed in various black listed sites. In general your server should not generate bounces so if you get lots you may find changing config settings can stop them. Note this only redirects bounces to non local recipients, so your users sending outgoing mail will still get their own bounce messages.

Syntax: `g_bounce_redirect` string

`g_bounce_reject` - Reject bounces by ip address from known dumb mail servers

Some mail servers (exchange) will accept email, then bounce it, this is now considered a 'crime' and will get your server black listed, so if you have surgemail running as a gateway for such servers you can tell it to reject any bounce that server is foolish enough to send you.

Syntax: `g_bounce_reject` string

`g_bounce_some_stop` - Disables locally generated bounces for partial message failure

This can decrease back scatter, but it has other bad effects, it can result in duplicate messages arriving

Syntax: `g_bounce_some_stop` bool

`g_bounce_suggest` - Send bounces to postmaster if spf cannot be verified

This may help stop black listing for backscatter while still alerting the sending domain admin that one of their users emails to your server bounced, You can specify a template file suggest.eml if you don't like the default message suggesting the postmaster add spf records for their domain

Syntax: `g_bounce_suggest` bool

`g_bounce_to` - Domains to treat as local and send bounces to

This setting makes bounces occur normally, the reason bounces are normally dropped for non local users is that they are almost always spam bouncing off another server due to forwarding settings, and as such sending a bounce email will get your server black listed, so we decided it was best to drop them by default since they are rarely useful. Turn this setting on at your own risk :-). Instead use `g_bounce_to` to list domains that it is safe to bounce to. e.g. `*@a.com,*@b.com`

Syntax: `g_bounce_to` string

`g_breakin_white` - Email addresses that can send from multiple ips

When a hacker guesses a password on your system they will often send outgoing spam to your server from multiple ip addresses, Surgemail detects this and emails the administrator when it occurs, use this setting to enable specific users who need to do this (this is very unusual though)

Syntax: `g_breakin_white` string

`g_bull_rule` - Post bulletins to this domain

Senders must be authenticated user that matches the sender, domain can be blank to send to all domains, the to field is the address you will send posts to, typically something like: `bulletins@your.domain.name`

Syntax: `g_bull_rule` to=string domain=string sender=string

`g_centipaid` - see [CentiPaid.htm](#)

Authentication server and port for CentiPaid.

Syntax: `g_centipaid` string

`g_cid_skip_to` - Skip CID score, good for lawyers etc

Some users will trigger CID matches due to the nature of their business (accountants/lawyers) for these people you may want to list them here. CID is content matching, usually scams which often use legal language.

Syntax: g_cid_skip_to string

See also: [g_friends_spam_score](#), [g_imap_spam_train](#), [g_spam_allow](#), [g_spam_allow_disable](#), [g_spam_allow_rbl](#), [g_spam_allow_msg](#), [g_spam_block_msg](#), [g_spam_allow_known](#), [g_spam_allow_recent](#), [g_spam_autotrain](#), [g_spam_block](#), [g_spam_block_gateway](#), [g_spam_check_auth](#), [g_spam_content_disable](#), [g_spam_body](#), [g_spam_body_url](#), [g_spam_body_more](#), [g_spam_folders](#), [g_spam_folders_show](#), [g_spam_flag](#), [g_spam_from_blacklist](#), [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spam_lang](#), [g_spam_probe](#), [g_spam_probe_unknown](#), [g_spam_probe_more](#), [g_spam_probe_whois](#), [g_spam_subject](#), [g_spam_subject_dom](#), [g_spam_subject_gateway](#), [g_spam_subject_word](#), [g_spam_userconfig](#), [g_spam_user_max](#), [g_spam_from_max](#), [g_spam_user_skip](#), [g_spam_bounce](#), [g_spam_bounce_text](#), [g_spam_bounce_all](#), [g_spam_bounce_trusted](#), [g_spam_cmd](#), [g_spam_cmd_if](#), [g_spam_cmd_skip](#), [g_spam_cmd_reject](#), [g_spam_vanish](#), [g_spam_vanish_all](#), [g_spam_info_hide](#), [g_spam_info](#), [g_spam_internal](#), [g_spam_noupdate](#), [g_spam_notrain](#), [g_spam_isspam_kind](#), [g_spam_isspam_ignore](#), [g_spam_aspam](#), [g_spam_poly](#), [g_spam_poly_disable](#), [g_spam_private](#), [g_spam_alias_any](#), [g_spam_url](#), [g_spam_catcher](#), [g_spam_char](#), [g_spam_notspam](#), [g_spam_hold_keep](#), [g_spam_hold_hide](#), [g_spam_header_trust_ip](#), [g_spam_share](#), [g_spam_status_hour](#), [g_spam_phishing](#)

g_comment - Management notes and comments about the server

This is a dummy setting that lets you store information in the ini file that will survive setting changes from the web admin tool.

Syntax: g_comment date=string name=string comment=string

g_con_perip - Connections per IP

Maximum number of connections allowed per IP address. Primarily this is used to prevent simple denial of service attacks where one user could otherwise use up all the channels your system can support and then do nothing with them.

Syntax: g_con_perip int

g_con_perip_except - Connections per IP exception

IP list of exception addresses to g_con_perip.

Syntax: g_con_perip_except string

g_con_persubnet - Maximum concurrent connections per subnet

Maximum number of concurrent connections per subnet. This limits concurrent connections from a sub net, great for automatically stopping professional spammers who use multiple addresses. A typical setting might be 20. Subnet is /24.

Syntax: g_con_persubnet int

g_convert_percent - Convert % signs top @ in recipient addresses

Some Spam tests send mail user%spamdomain.com@localdomain.com to see if a server is an open relay. If a default address is set up for the local domain this will be delivered to this local address and the test assumes the mail server is an open relay. This setting prevents this.

Syntax: g_convert_percent bool

g_country_ip - Tag messages with country of origin

Downloads a ip to country database and then adds a header based on that to each message to show where it came from. This file IpToCountry.csv should appear in your surgemail home directory after enabling this setting (restart surgemail too), if the file doesn't appear you can download it via <http://netwinsite.com/surgemail/IpToCountry.csv>

Syntax: g_country_ip bool

See also: [spam_strip](#), [spam_block](#), [spam_noblock](#), [g_aspam_headers](#), [g_aspam_need_ip](#), [g_black_isspam](#),

[g_friends_spam_score](#), [g_imap_spam_train](#), [g_report_spam](#), [g_report_notspam](#), [g_spam_allow](#), [g_spam_allow_disable](#), [g_spam_allow_rbl](#), [g_spam_allow_msg](#), [g_spam_block_msg](#), [g_spam_allow_known](#), [g_spam_allow_recent](#), [g_spam_autotrain](#), [g_spam_block](#), [g_spam_block_gateway](#), [g_spam_check_auth](#), [g_spam_content_disable](#), [g_spam_body](#), [g_spam_body_url](#), [g_spam_body_more](#), [g_spam_folders](#), [g_spam_folders_show](#), [g_spam_flag](#), [g_spam_from_blacklist](#), [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spam_lang](#), [g_spam_probe](#), [g_spam_probe_unknown](#), [g_spam_probe_more](#), [g_spam_probe_whois](#), [g_spam_subject](#), [g_spam_subject_dom](#), [g_spam_subject_gateway](#), [g_spam_subject_word](#), [g_spam_userconfig](#), [g_spam_user_max](#), [g_spam_from_max](#), [g_spam_user_skip](#), [g_spam_bounce](#), [g_spam_bounce_text](#), [g_spam_bounce_all](#), [g_spam_bounce_trusted](#), [g_spam_cmd](#), [g_spam_cmd_if](#), [g_spam_cmd_skip](#), [g_spam_cmd_reject](#), [g_spam_vanish](#), [g_spam_vanish_all](#), [g_spam_info_hide](#), [g_spam_info](#), [g_spam_internal](#), [g_spam_noupdate](#), [g_spam_notrain](#), [g_spam_isspam_kind](#), [g_spam_isspam_ignore](#), [g_spam_aspam](#), [g_spam_poly](#), [g_spam_poly_disable](#), [g_spam_private](#), [g_spam_alias_any](#), [g_spam_url](#), [g_spam_catcher](#), [g_spam_char](#), [g_spam_notspam](#), [g_spam_hold_keep](#), [g_spam_hold_hide](#), [g_spam_header_trust_ip](#), [g_spam_share](#), [g_spam_status_hour](#), [g_spam_phishing](#), [g_spamdetect_always](#)

g_create_allow - List of characters allowed in usernames/passwords

Defaults to A-Za-z0-9_- meaning usernames/password may contain letters, numbers, -, _ and . and nothing else.

Syntax: g_create_allow string

g_create_allow_pass - List of characters allowed in passwords

Setting overriding g_create_allow just for passwords.

Syntax: g_create_allow_pass string

g_create_apply - List of user groups to apply create_* settings for.

This setting allows you to apply create_* settings to domain admin accounts. Specify g_access_group names and domain admins in these groups will have create_* settings applied to them when adding users in the domain admin interface.

Syntax: g_create_apply string

g_create_badnames - List of illegal usernames

Comma separated list of illegal usernames, may contain wild cards, if username contains part of a non-wild card or matches a wildcard it is disallowed.

Syntax: g_create_badnames string

g_create_cleanup - Cleanup existing data before adding a user

This causes a delete to be actioned for a user before/as they are created. This ensures the new user does not end up with any files, on any mailing lists, with any aliases etc from a previous user of the same name/address. If you delete users from the authent database directly i.e. not using the surgemail web admin or calling 'tellmail delete_user' then this setting will cleanup the users files when their address is re-used.

Syntax: g_create_cleanup bool

g_create_dictionary - File containing dictionary words to compare passwords to

Text file containing one word per line, passwords are compared to all words longer than 4 characters in this file, if a username or password contains a word in this file it is not allowed. Only takes effect if g_create_strict is checked.

Syntax: g_create_dictionary string

g_create_pass_length - Limit the length of user passwords

This is applied during user self creation and when users change passwords. Set admin to true to restrict the domain and global admin also.

Syntax: g_create_pass_length min=int max=int admin=bool

g_create_record_ip - Causes surgemail to store ipnum in the authent database

This setting has no further documentation currently available

Syntax: g_create_record_ip bool

g_create_strict - Whether to apply strict rules to usernames/passwords

Checking this causes surgemail to check passwords do not contain words longer than 4 characters from g_create_dictionary as well as requiring the password to be 6+ characters, and usernames/passwords to contain more than 1 character.

Syntax: g_create_strict bool

g_create_strict_admin - Enforce strict rules for admins too, set g_create_strict AS WELL!!

This setting has no further documentation currently available

Syntax: g_create_strict_admin bool

g_create_user_length - Limit the length of usernames

This is applied during user self creation. Set admin to true to restrict the domain and global admin also.

Syntax: g_create_user_length min=int max=int admin=bool

g_dbabble_links - Add web links to DBabble from other web interfaces (and vice versa)

This causes links to appear in the DBabble interface to switch to using WebMail (and SurgePlus if you have the g_surgeplus_links setting on).

Syntax: g_dbabble_links bool

See also: [g_dbabble_smtp_port](#), [g_dbabble_smtp_prefix](#)

g_dbabble_smtp_port - DBabble SMTP port (do not manually change this setting - it should be set from the DBabble section of the web admin interface only)

This setting specifies the port that DBabble listens on. DBabble looks at surgemail.ini and if it sees this setting, overrides it's own setting with this value. When you save changes to this setting from within the SurgeMail DBabble admin interface, SurgeMail automatically sets appropriate values for the g_redirect_iflocal and g_gateway settings.

Syntax: g_dbabble_smtp_port int

See also: [g_dbabble_smtp_prefix](#), [g_dbabble_links](#)

g_dbabble_smtp_prefix - DBabble SMTP prefix (do not manually change this setting - it should be set from the DBabble section of the web admin interface only)

This setting is used in conjunction with the dbabble_smtp_port setting to forward all mail with the specified prefix on to DBabble.

Syntax: g_dbabble_smtp_prefix string

See also: [g_dbabble_smtp_port](#), [g_dbabble_links](#)

g_debug_crt - Some CRT debugging on windows, do not use

This setting has no further documentation currently available

Syntax: g_debug_crt bool

g_debug_free - Check free memory isn't corrupted - slows performance slightly
This is for tracking a particular bug, not for general use

Syntax: **g_debug_free** bool

g_debug_ini - Debugging, don't use this
This is a temp setting used for testing

Syntax: **g_debug_ini** bool

g_debug_vanished - Name of file to check for, if file vanishes, crash
This is for tracking a particular bug, not for general use

Syntax: **g_debug_vanished** string

g_delete_exclude - Field and value that excludes an account from **g_delete_user_after**
If the authent response includes this field/value pair then the user account will not expire

Syntax: **g_delete_exclude** field=string value=string

Example: field="noexpire" value="true"

See also: [g_acctlog_authonly](#), [g_authent_always](#), [g_authent_any](#), [g_authent_allow_badascii](#),
[g_authent_prefix_sep](#), [g_authent_process](#), [g_authent_cachelife](#), [g_authent_cachebad](#),
[g_authent_cachesize](#), [g_authent_domain](#), [g_authent_encrypt_key](#), [g_authent_number](#), [g_authent_info](#),
[g_authent_info_grp](#), [g_authent_ip](#), [g_authent_path_broken](#), [g_authent_single](#), [g_authent_spaces](#),
[g_authent_strip_domain](#), [g_authent_restart](#), [g_authent_logall](#), [g_authent_fwdfile](#), [g_authent_timeout](#),
[g_authent_last_login](#), [g_auth_hide](#), [g_auth_norelay](#), [g_auth_skipgateway](#)

g_delete_user_after - Number of days an account can remain unread before it is deleted

DO NOT USE THIS SETTING IN A MIRROR/CLUSTER SETUP

Number of days an account can remain unread before it is deleted. This setting cannot be used on an **authent_domain** FALSE domain unless it has a [prefix](#) setting.

Syntax: **g_delete_user_after** int

g_delete_user_mode - What to do when an account is unread

DO NOT USE THIS SETTING IN A MIRROR/CLUSTER SETUP

You can set this to "file" or "suspend". "file" causes accounts to be written to the **users_delete.rec** file, which you can action by running "tellmail delete_user FILE" or "tellmail delete_user FILE **users_delete.rec**" (optionally specify the file). "suspend" causes accounts to be suspend, it does this by setting the field and value specified in the [g_delete_user_suspend](#) setting.

If this setting is blank the default is to use 'file' mode, accounts are NEVER deleted automatically except in the very oldest versions of surgemail (before version 3)

Syntax: **g_delete_user_mode** string

g_delete_user_suspend - If suspending an unread account set this field/value

DO NOT USE THIS SETTING IN A MIRROR/CLUSTER SETUP

Set the field and value to use when suspending an account due to [g_delete_user_after](#) and the [g_delete_user_mode](#) "suspend" settings.

Syntax: **g_delete_user_suspend** field=string value=string

g_deny - Deny users from some IP ranges

Block known spammers etc by IP address. You can use wild cards and 'not' signs, e.g. "!* ,127.* ,10.*"

Syntax: **g_deny** string

g_deny_msg - Deny message

Message to give to users who are disconnected due to the above 'deny' setting.

Syntax: g_deny_msg string

g_deny_smtp - Deny SMTP based on IP address

Block users from some IP ranges connecting to SMTP only.

Syntax: g_deny_smtp string

g_disable_exclude - Field and value that excludes an account from g_disable_smtp_after

If the authent response includes this field/value pair then the user account will not be disabled from receiving messages

Syntax: g_disable_exclude field=string value=string

Example: field="noexpire" value="true"

See also: [g_disable_smtp_after](#)

g_disable_skip - Ip address of senders to accept email from even if user account is disabled due to g_disable_smtp_after

Useful to ensure delivery for important company notices

Syntax: g_disable_skip string

g_disable_smtp_after - Number of days an account can remain unread before delivery is disabled

DO NOT USE THIS SETTING IN A MIRROR/CLUSTER SETUP

Number of days an account can remain unread before delivery is disabled.

Syntax: g_disable_smtp_after int

g_disable_surgeplus - Disable SurgePlus Calendar and File Sharing client

Disable users from logging in using the SurgePlus Calendar and File Sharing client. See [SurgePlus](#)

Syntax: g_disable_surgeplus bool

See also: [old_xfile](#), [xfile_url](#), [disable_surgeplus](#), [surgeplus_pop_server_name](#), [surgeplus_smtp_server_name](#), [g_xfile_allow](#), [g_surgeplus_links](#), [g_disable_surgeplus_updates](#), [g_surgeplus_log_level](#), [g_surgeplus_port](#), [g_surgeplus_secure_port](#), [g_surgeplus_web_port](#), [g_surgeplus_web_url](#), [g_surgeplus_hide_client_downloads](#), [g_surgeplus_pop_server_name](#), [g_surgeplus_smtp_server_name](#), [g_surgeplus_delay_tell_upgrade](#), [g_surgeplus_delay_tell_upgrade_exempt](#), [g_surgeplus_online](#)

g_disable_surgeplus_updates - Disable automated downloading of new versions of SurgePlus client from netwinsite.com

New versions of the SurgePlus client are automatically downloaded from netwinsite.com and made available for download from your server by your users. See [SurgePlus](#)

Syntax: g_disable_surgeplus_updates bool

See also: [disable_surgeplus](#), [g_disable_surgeplus](#), [g_surgeplus_delay_tell_upgrade](#), [g_surgeplus_delay_tell_upgrade_exempt](#)

g_diskio_abort - Shutdown if diskIO failure on queue files

Intended to make server die rather than to pretend to keep running when a major disk fault has occurred

Syntax: g_diskio_abort bool

g_dlist_nolocal - Remove add local button from mailing lists

Prevents address harvesting etc by users - strongly recommended on public servers, not necessary on

small or private servers

Syntax: `g_dlist_nolocal` bool

g_dlist_nostart - Disable dlist

If set disable (do not attempt to start) dlist for DMail compatibility mode..

Syntax: `g_dlist_nostart` bool

g_dlist_path - Path for dlist

DList Path normally defaults to `$g_home/dlist`.

Syntax: `g_dlist_path` string

g_dns_cache_size - Set size of forward dns cache, default 7000

Best not to change this normally

Syntax: `g_dns_cache_size` int

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_host - DNS host(s) for MX lookups

This setting can normally be left blank as the mail server will find your system DNS settings. However, you can specify one or more DNS servers for the mail server to use instead to lookup names.

DNS lookups are cached to disk so SurgeMail will generally continue to work even if your dns server is temporarily unavailable.

Test your dns server with this command. If working it should return two ip addresses for that domain.

```
tellmail dns_test "netwinsite.com"
```

Prior to SurgeMail 2.0h dns lookups were done using tcp instead of udp, they are now down with UDP unless the response exceeds UDP packet size (as per RFC).

NOTE: All dns servers listed in this setting must be fully recursive, a non recursive dns server will create many dns lookup failures!

Syntax: `g_dns_host` string

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_match_msg - Message for stamp or bounce if forward and reverse lookup don't match

The message given to the user when the forwar/reverse dns lookup doesn't match

Syntax: `g_dns_match_msg` string

Example: "Sorry your ip address doesn't translate into a name that translates into your ip address"

See also: [g_dns_paranoid](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_nlookup - Concurrent MX lookups

Concurrent DNS lookups to send to DNS server (Default=20) (not used after version 2.0h)

Syntax: `g_dns_nlookup` int

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#),

[g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_nocache - Disables DNS cache for spf lookups (20 minute life)

This setting disables the small cache used for SPF lookups to improve performance.

Syntax: **g_dns_nocache** bool

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_noptr - Set to reject or retry, for ip addresses with no reverse dns entry (rdns)

If the ip number of a connecting user has no associated name in the reverse dns database then the connection is rejected or told to retry later.

Syntax: **g_dns_noptr** string

Example: "retry"

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_noptr_msg - Message for stamp or bounce if DNS lookup fails on ip address

See short description.

Syntax: **g_dns_noptr_msg** string

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_noptr_skip - Skip RDNS for these ip addresses

This is an over-ride for local addresses which you trust.

Syntax: **g_dns_noptr_skip** string

Example: "retry"

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_paranoid - Compare sender forward and reverse dns lookup and see if they match

Does a forward DNS lookup on the sender's domain and matches this with a reverse lookup of the senders IP address. If these do not match the message is either bounced or stamped with the header "X-DNS-Paranoid: <explanation>". Valid values for this field are "STAMP", "RETRY" and "REJECT".

STAMP = Add the X-DNS-Paranoid header if it fails

RETRY = Bounce the message with a 450 error. (so if the failure was temporary the sending server will retry)

REJECT = Bounce the message with a 550 error

Set [g_dns_lookup_msg](#) or [g_dns_match_msg](#) to define the reject/stamp strings respectively.

g_dns_require - Require reverse DNS names match

Require MAIL FROM header to match the reverse dns lookup based of the sender based on the sender's IP.

eg. from=*@hotmail.com hosts=hotmail.com

Syntax: **g_dns_require** string

See also: [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_require - Require MAIL FROM header matches senders ip reverse dns

This setting predates SPF which does the same sort of thing on a grander scale, no longer needed.

Syntax: g_dns_require from=string hosts=string

Example: from=*@hotmail.com hosts=hotmail.com

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_system - Use system code to do reverse lookups

If all channels hang in a state 'lookup' then turn this off so it will use the surgemail code for reverse dns lookups. This setting used to be g_dns_lookup and had the opposite meaning, we reversed it because the system dns code was faulty so often

Syntax: g_dns_system bool

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_dns_translate - If mx response is x.x.x.x translate to y.y.y.y:port

Useful for translating ip numbers inside a local intranet and doing other fancy routing of various sorts.

Syntax: g_dns_translate from=string to=string

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_domadmin_utoken_expire - Length of time a domain admin login token is valid for in seconds

Default unit is seconds. You can specify units e.g. 3 minutes, 10 hours etc...

Syntax: g_domadmin_utoken_expire int

g_domadmin_utoken_idle - Length of time a domain admin login token may remain idle for

This setting has no further documentation currently available

Syntax: g_domadmin_utoken_idle int

g_domain_default - Default domain when POP/IMAP user does not specify one

This is probably not what you think it is, generally the 'first' domain in surgemail.ini is used in this situation, but in some instances, when using domuser.dat for example to translate users back to virtual domains, you will want the default domain to be a 'generic' made up domain that doesn't really exist.

For example lets say you have users fred@a.com, bob@b.com, then in domusers.dat you have

```
fred@a.com fred@a.com
bob@b.com bob@b.com
bob@xxx bob@b.com
fred@xxx fred@a.com
```

And the result is that users who login to pop as bob or fred, will be correctly mapped to the correct virtual domain user even though the actual domain is different in those two cases.

Clear as mud I expect?

Syntax: g_domain_default string

g_domain_list_max - Maximum number of domains to list at once

Maximum number of domains to list at once in the admin user interface.

Syntax: g_domain_list_max int

g_domain_separator - Separator characters for virtual POP

For POP logins where your virtual domain is NOT distinguished by IP address users can login with 'user@domain' or user/domain.name etc and the mail server will pickup the domain name correctly. By default only 'user@domain.name' is accepted unless this setting is used which can be useful for brain dead mail clients which don't allow the user to specify 'user@domain.name' as the username eg:

g_domain_separator "/"

Syntax: g_domain_separator string

g_domainkeys_check - Check incoming DomainKeys signatures (beta may be unstable)

See domainkeys.htm

Syntax: g_domainkeys_check bool

g_domainkeys_headers - List which headers to sign

This will help get the message through gateways without breaking the signature, try a single header, e.g. from

Syntax: g_domainkeys_headers string

g_domainkeys_only - Domains to sign for outgoing email

Normally all local domains are signed, but if this setting exists then it is used instead so you must list local domains as well as non local ones you want to sign messages for. G_domainkeys_sign must also be set to true!

Syntax: g_domainkeys_only string

g_domainkeys_selector - Policy name for your server (used creating dns entry for domainkeys)

This defines the dns entry name for your policy record and public key entry in your dns. See domainkeys.htm for details

Syntax: g_domainkeys_selector string

g_domainkeys_sign - Sign outgoing messages (create a key first using web admin)

To turn off domainkeys for some domains see the per domain setting, domainkeys_disable. See domainkeys.htm for more info.

Syntax: g_domainkeys_sign bool

g_domuser_file - Domain users to thousands of virtual domains easily

Specifies a file which contains lines that translate an email address to the username that should be looked up in the database. This file can contain a domain name not previously specified in surgemail.ini allowing you to create unique sub-domain addresses. eg:

g_domuser_file "c:\surgemail\domuser.dat"

Example entries...

*@domain.com postmaster@domain.com

userA@domain.com userB@domain.com

firstname@lastname.domain.com firstname@lastname.domain.com

Syntax: g_domuser_file string

g_dotlock_minutes - NFS lock waits

Minutes to wait for nfs lock file, default 20 minutes.

Syntax: g_dotlock_minutes int

g_dotstuff_fix - Convert the way mail is stored on disk from dotstuffed to non dot stuffed (beta)

In the dotstuffed format any attachments that have content (in encoded format) starting with a . get corrupted, as all single '.' characters at the start of a line are converted to '..'. This is only very seldomly an issue as encoded text doesn't usually have . characters. This feature can only be enabled and still need further production level testing to make sure there are no side effects... so if you play with it consider yourself adequately warned :-)

Syntax: g_dotstuff_fix bool

g_download - Fetch an http file and do an ini reload

Can be used with g_include to have settings fetched from a central location, the file is fetched once an hour.

Syntax: g_download url=string user=string pass=string local=string

g_drop_use_len - Use the content-len header for drop file processing

For use on Solaris when using sendmail for incoming mail delivery.

Syntax: g_drop_use_len bool

g_dsn_enable - Enable DSN (Delivery Status Notification) esmtp extension.

Not recommended. Delivery Status Notification is used by spammers to find addresses to spam to.

Syntax: g_dsn_enable bool

See also: [g_dsn_nofinal](#)

g_dsn_nofinal - Try not to show real final recipients but just original recipients

This setting helps hide internal addresses in bounce messages (after forwarding etc). Not recommended.

Syntax: g_dsn_nofinal bool

See also: [g_dsn_enable](#)

g_ehlo_simple - Ip addresses to give simple ehlo response to

This is a debugging setting, do not use.

Syntax: g_ehlo_simple string

g_encrypt_expire - Days to keep encrypted messages, default 60

When a message is sent via encryption it is deleted after this many days

Syntax: g_encrypt_expire int

g_encrypt_inline - Use INLINE method by default

Sets the default encryption method when a rule does not apply

Syntax: g_encrypt_inline bool

g_encrypt_max - Max encrypted per day server wide

Server wide limit to prevent abuse (or accidental over use)

Syntax: g_encrypt_max int

g_encrypt_path - Path to encrypted files, this is not supported when mirroring!

DO NOT USE

Syntax: g_encrypt_path string

**g_encrypt_pw_host - Central host for encryption password storage
DO NOT USE**

Syntax: g_encrypt_pw_host string

**g_encrypt_pw_key - Central host password key
DO NOT USE**

Syntax: g_encrypt_pw_key string

g_encrypt_reply_plain - Send plain message for local replies

By default a reply to a local user is also encrypted this makes it not encrypt the reply as user should be reading the message via SSL so the data is secure anyway.

Syntax: g_encrypt_reply_plain bool

g_encrypt_ssl_force - Require ssl on incoming encrypted messages

When a message is going to be encrypted this setting ensures it is sent from the user to the server via SSL

Syntax: g_encrypt_ssl_force bool

g_encrypt_ssl_noforce - Exceptions, e.g. surgeweb or localhost

When a message is going to be encrypted this setting ensures it is sent from the user to the server via SSL

Syntax: g_encrypt_ssl_noforce string

g_encrypt_surgeweb_show - Show SurgeVault in SurgeWeb

Enables the display of surgevault encryption in the surgeweb interface (can be modified using encrypt_hide on surgeweb customisation page)

Syntax: g_encrypt_surgeweb_show bool

g_enotify_from - From address to use in email notification messages

This setting has no further documentation currently available

Syntax: g_enotify_from string

g_eof_fix_off - Turns off auto stripping of control+Z

These characters can break some mail clients and should not appear in normal emails

Syntax: g_eof_fix_off bool

g_error_xlate - Change error messages

If wild card string matches smtp response code, then replace with 'to' response code, use %1 to replace the first wild card match etc...

Syntax: g_error_xlate was=string to=string

g_expire_every - Only expire spool once every 'n' days

Reduce load spent expiring old messages.

Syntax: g_expire_every int

See also: [expire_age](#), [expire_size](#), [expire_rule](#), [g_encrypt_expire](#), [g_expire_trash](#), [g_expire_silent](#), [g_expire_warning](#), [g_user_utoken_expire](#), [g_admin_utoken_expire](#), [g_domadmin_utoken_expire](#)

g_expire_silent - Don't send users emails telling them what was expired.

Some users get upset when they find messages have expired, this setting makes the expiration silent so the users don't even notice. I think this is a bit nuts myself but some admins prefer it

Syntax: g_expire_silent bool

See also: [expire_age](#), [expire_size](#), [expire_rule](#), [g_encrypt_expire](#), [g_expire_trash](#), [g_expire_every](#), [g_expire_warning](#), [g_user_utoken_expire](#), [g_admin_utoken_expire](#), [g_domadmin_utoken_expire](#)

g_expire_trash - Expire any messages found in trash folders

Expires any messages more than 7 days old found in the 'trash' folder.

Syntax: g_expire_trash bool

See also: [expire_age](#), [expire_size](#), [expire_rule](#), [g_encrypt_expire](#), [g_expire_silent](#), [g_expire_every](#), [g_expire_warning](#), [g_user_utoken_expire](#), [g_admin_utoken_expire](#), [g_domadmin_utoken_expire](#)

g_expire_warning - Give warning 'n' days before deleting each file

This will help warn users before a file is actually deleted.

Syntax: g_expire_warning int

See also: [expire_age](#), [expire_size](#), [expire_rule](#), [g_encrypt_expire](#), [g_expire_trash](#), [g_expire_silent](#), [g_expire_every](#), [g_user_utoken_expire](#), [g_admin_utoken_expire](#), [g_domadmin_utoken_expire](#)

g_external_ip_disable - Do not add X-External-IP header

Please note you may wish to remove x_originating_ip true from webmail.ini as well

Syntax: g_external_ip_disable bool

g_fallback - Fallback address

Default address for all local domains. If a local delivery is not to any valid user Emails will be delivered to this address. There is also a per domain default.

We want to stress that this is a dangerous setting, you use at your own peril.

Spammers will turn up to your server and test sending to accounts, they will just run through a dictionary of names, with a fallback setting you will be telling the spammer that all these accounts exist. The spammer will then deliver spam to these addresses in volumes that can cripple a server almost.

Syntax: g_fallback string

g_fallback_relay_if_exists - Use FALLBACK_RELAY if not logged in but user exists
(OLD_POPOST_CREATEUSER_DISABLE)

This can be used to relay users where you have a user database that can be checked on the front end system directly (odbcauth, tcpauth, etc)

Syntax: g_fallback_relay_if_exists bool

See also: [surgevall](#), [surgevall_auth](#), [surgevall_local_too](#), [surgevall_options](#), [surgevall_capa_local](#), [g_surgevall_split](#)

g_filter_max - Max size of messages to send through the filter pipe

Messages over this size (in bytes) are skipped. default = no limit

Syntax: g_filter_max int

g_filter_n - Number of filters to run simultaneously

Default is 20, when this limit is reached the incoming thread waits a few seconds then skips the filter if necessary, this is intended to prevent a log jam/melt down effect.

Syntax: `g_filter_n` int

g_filter_pipe - Filter pipe allowing external message processing

This allows external applications to filter and modify incoming messages. Example: Integration with Spam Assassin (on UNIX) could be achieved as follows:

`g_filter_pipe "/usr/local/bin/spamassassin -P"`

it expects a normal unix 'filter' so, read the message on 'stdin' and write the identical (or modified) message to 'stdout'.

The input will be 'crlf' terminated and so should the output file.

That's all you can do with this mechanism, if you want to bounce the message or flag it as spam you 'add' a header and then use something in surgemail to detect and act on the header you've added (mfilter)

Syntax: `g_filter_pipe` string

g_filter_pipe_noauth - Skip for auth users Skip for authenticated users

Syntax: `g_filter_pipe_noauth` bool

g_filter_pipe_skip - Skip filter if ip matches this

Set this for local servers that don't need filtering, e.g. mailing list servers, local trusted robots.

Syntax: `g_filter_pipe_skip` string

g_filter_timeout - Filter pipe timeout

Filter timeout (`g_filter_pipe`) in seconds, default is 360.

Syntax: `g_filter_timeout` int

g_fix_crcrlf - Fix email messages containing crcrlf for line termination

This is best not used, it's best to fix the faulty email application, results are not gauranteed.

Syntax: `g_fix_crcrlf` bool

g_fix_imap_If - During IMAP import fix email messages containing If

This is best not used, it's best to fix the faulty email server, results are not gauranteed.

Syntax: `g_fix_imap_If` bool

g_footer_file - Footer file

Footer file which is appended to all plain text mail messages.

Syntax: `g_footer_file` string

g_footer_html - Footer file (HTML mail)

Footer file which is appended to all HTML mail messages.

Syntax: `g_footer_html` string

g_footer_notfound - Only add footer if footer is not in message already

This works by examining the message contents to try and find part of the footer.

Syntax: `g_footer_notfound` bool

g_footer_send - Footer file (outbound only)

Plain text footer file which is appended to all outbound mail messages only.

Syntax: `g_footer_send` string

g_footer_sendonly - Enable outbound footer

Add g_footer_send to all messages when sending to non local users.

Syntax: g_footer_sendonly bool

g_footer_skip - Skip footers for these users

This skips the footer for matching users (e.g. cell phones etc)

Syntax: g_footer_skip string

g_footer_trusted - Only add footers if sender is trusted

This prevents the footer from being added for a message that pretends to come from your domain.

Syntax: g_footer_trusted bool

g_forward_attach - When late forwarding send as attachment to these domains

Useful with hotmail.com, aol.com etc so that forwarded messages are not mistaken for spam

Syntax: g_forward_attach string

g_forward_illegal - Prevents users setting forward rules to certain addresses

Syntax: g_forward_illegal to="address" apply="user type "

This setting allows you to specify some addresses as being illegal for certain users. This stops users setting up forwarding rules to these addresses. They can still send mail to these addresses manually with their email client. These rules ONLY apply to non local domains.

Some examples:

If you want to stop your users setting up forward rules that redirect to aol.com.

g_forward_illegal to="*@aol.com" apply="user"

If you want to stop your users setting a forward to all domains except aol.com

g_forward_illegal to="*;!*@aol.com" apply="user"

Stop domain admins sending to aol.com

g_forward_illegal to="*@aol.com" apply="domadmin"

Stop admins sending to netwinsite.com

g_forward_illegal to="*@netwinsite.com" apply="admin"

Syntax: g_forward_illegal to=string apply=string

g_forward_oops - Internal testing setting, not for general use sorry

Testing setting, please do not use.

Syntax: g_forward_oops string

g_friends_add_trusted - Add to friends list when if sender is trusted

This is useful if senders are not using smtp auth but you still want friends to be added, typically used with surgewall...

Syntax: g_friends_add_trusted bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#),

[g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_allow_spf - Allow all email through as if it was a friend during temporary allow
The user click on a button to disable friends for a few hours, during this time all messages will get treated as a friend and thus bypass SPF too.

Syntax: **g_friends_allow_spf** bool

g_friends_always - Always use friends list.

This enables the "Add all outgoing email addresses to list" feature and always checks incoming messages against the friends list so that SurgeMail can correctly tag or filter it.

Syntax: **g_friends_always** bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_at_rcpt - Whether to check users friends list at rcpt stage

This setting is automatically added/removed by the web admin when global friends defaults are configured. It allows us to check friends at rcpt stage without paying a disk access cost for non-friends users.

Syntax: **g_friends_at_rcpt** bool

g_friends_bounce_rej - Reject blank return path as friends failures

This setting has no further documentation currently available

Syntax: **g_friends_bounce_rej** bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_bounce_second - Bounce the next time the user sends a message if waiting for confirm still
This can make it clearer that email is not getting through to the destination

Syntax: **g_friends_bounce_second** bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_byweb - Perform confirmation of pending via the web.

This gives a confirmation url to the sender in the friends confirmation message and disables confirmation by replying. The url gives a page with a verification image, the sender types the number seen and releases their message.

Syntax: `g_friends_byweb` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

`g_friends_check_spf` - Disable friends bounces if SPF headers missing/failed to avoid backscatter. If the incoming message may be forged it will bounce messages using an smtp error code to deny delivery but it will allow any real sender to bypass this. This settings is good if spamcop block your domain for sending friends challenges as it cuts down on the number of such messages. This avoids backscatter

Syntax: `g_friends_check_spf` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

`g_friends_confirm_debug` - Log sucessful friends confirmation responses

This enables us to examine suspect replies to friends confirmations for indications that they were sent by spammers or mail robots.

Syntax: `g_friends_confirm_debug` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

`g_friends_confirm_subject` - String to use as the subject of a friends confirmation email

String to use as the subject of a friends confirmation email. Defaults to: *"Please reply to ||confirm|| message and allow delivery"*. This value **must** contain the text `||confirm||`, this text is replaced by the unique message id that allows SurgeMail to find the message to release eg. `confirm(1150419513.1880_1180.domain)`. It is also advisable to place the `||confirm||` near the start of the string as some clients will truncate long subjects and any truncation of the `||confirm||` value will result in failure to release the message.

Syntax: `g_friends_confirm_subject` string

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#)

[g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_default_autoadd - Default auto addition when sending (recommended)

This setting has no further documentation currently available

Syntax: `g_friends_default_autoadd bool`

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_default_mode - Default friends mode (Recommended 'list')

This setting has no further documentation currently available

Syntax: `g_friends_default_mode string`

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_ignore - List of addresses considered friends for all users on the system

List of addresses considered friends for all users on the system eg: the system manager email address

Syntax: `g_friends_ignore string`

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_ignore_trusted - If from trusted ip still apply friends

Useful when you have a gateway that is sending to surgemail

Syntax: `g_friends_ignore_trusted bool`

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_latest_headers - Friends system re-read message headers

Causes friends to re-read message headers, allowing rules based on headers added during delivery

Syntax: `g_friends_latest_headers` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_name - What to call the friends system

This specifies what to call the friends system when referring to it on web pages and in email to our users, you can call it whatever you like

Syntax: `g_friends_name` string

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_old_status_email - Use older status email & processing

Use `status.eml` instead of `status.html.eml`

Syntax: `g_friends_old_status_email` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_only - Friends system

An anti-spam feature which screens incoming mail to ensure it comes from a human. For incoming mail from unknown addresses a message is sent to this person requesting them to reply to confirm they are human and the original message will be delivered. [See this page for more details.](#)

Syntax: `g_friends_only` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_pending_keep - Time to keep friend pending messages

How long to store users friends pending messages before deleting them (days)

Syntax: `g_friends_pending_keep` int

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#),

[g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_pending_name - The imap name of the friends_pending (and spam store) quarantine folder - should match surgeweb imap_spam_folder - default is 'Friends Pending'

This shouldn't be changed unless this feature has not been used before as it will confuse your users. Any matching folder the user has of the same name will become invisible. So at least make it something other than simply Spam!!

Syntax: g_friends_pending_name string

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_pending_vanish - Enable auto-vanish of pending messages on confirmation bounce

When a bounce for a confirmation message is received we vanish it, this setting will also delete the original message.

Syntax: g_friends_pending_vanish bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_release_wash - Clean any subject marking (ie stars) when releasing/allowing

This setting has no further documentation currently available

Syntax: g_friends_release_wash bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_rotate - Rotate user level log file, default 30k

Set log size, the log is also rotated when a friends report email is sent (if configured)

Syntax: g_friends_rotate int

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#),

[g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_short - In friends web release addresses use a shortened url

This can help prevent urls being wrapped by dumb email clients, this feature requires an updated template file na_rel_link.htm with the 'f' hidden input field, e.g. input type="hidden" name="f" value="||f||"

Syntax: **g_friends_short** bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_silent - Disable friends responses to users

This setting is to simply disable the confirm emails, not generally recommended as this makes friends a bit pointless.

Syntax: **g_friends_silent** bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_skip_ip - List of ip addresses considered friends for all users on the system

This setting has no further documentation currently available

Syntax: **g_friends_skip_ip** string

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_friends_spam_score - Default spam score for friends spam scoring (surgeweb defaults likely=4 or almost_certain=10)

This setting has no further documentation currently available

Syntax: **g_friends_spam_score** int

g_friends_spf_fail_bounce - Bounce SPF failures, do not send friends confirmations (Not recommended)

The default behaviour is to only send confirmations if SPF checks pass, if they fail friends checking is skipped, no confirmation request is sent and the email is not blocked by friends.

Syntax: `g_friends_spf_fail_bounce` bool

g_friends_url - Specify full url for friends release `http://domain.name:port`
Normally the default will work.

Syntax: `g_friends_url` string

g_from_allow - From header allow

From headers to allow bypassing the `g_from*` checks. e.g. `"**@x.y.com,*@b.com,fred@bb.com"`

Syntax: `g_from_allow` string

g_from_allow_ip - IP addresses to bypass local from check

This setting has no further documentation currently available

Syntax: `g_from_allow_ip` string

g_from_allow_to - destination user to bypass local from check

This setting has no further documentation currently available

Syntax: `g_from_allow_to` string

g_from_bl - Domain Based Blacklist Zones, lookups FROM domain in dns

The 'from' domain is checked against the specified RBL which must be a special 'FROM' based rbl which lists spammers by from address. Most spammers fake from addresses so this is a fairly marginally useful method.

Syntax: `g_from_bl` name=string stamp=string

See also: [g_honeypot_rbl](#), [g_myrb1 testing](#), [g_myrb1 to](#), [g_myrb1 store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl reject](#), [g_surbl whois](#), [g_surbl skip](#), [g_surbl skip ip](#)

g_from_bounce - Bounce if from is probably faked

Bounce if from address is probably faked.

This check is activated for any mail with a local domain in the from address but not using SMTP authentication, relay allow IP address or spam allow IP address.

Syntax: `g_from_bounce` bool

g_from_check - Check from matches valid local domain

Check from domains match valid local domains if user is authenticated, or `g_from_allow`.

Should be used with `g_from_bounce "true"` which basically forces them to authenticate and then makes this setting work properly.

Syntax: `g_from_check` bool

g_from_domain - Default domain for from envelope

Fixes the 'from' envelope if the email client failed to specify a domain name, this doesn't fix the from header currently but we may change that in future!

Syntax: `g_from_domain` string

g_from_exact - Check from matches authenticated user

Check from matches authenticated user. If user is not authenticated the setting is skipped.

Should be used with `g_from_bounce` "true" which basically forces them to authenticate and then makes this setting work properly.

Syntax: `g_from_exact` bool

g_from_header - From header used in delivery bounces

From header used in delivery bounces.

Syntax: `g_from_header` string

g_from_must_exist - Require local from addresses to exist or reject mail

Can be useful in blocking dumb spam robots

Syntax: `g_from_must_exist` bool

g_from_noforge - If envelope or from is local domain then the other must be too

This can prevent many common forms of forgery, this will bounce some real email, so probably better to use the `noforgeme` setting instead. One of the settings to prevent forgery

Syntax: `g_from_noforge` bool

g_from_noforge_some - If from matches this then from/envelope must match

Prevent forgeries of important local addresses, e.g. `*support*`

Syntax: `g_from_noforge_some` string

g_from_noforgeme - If to==from then from and env from must match

This can prevent many common forms of forgery, this is safer than the `noforge` setting above, and generally almost as effective. One of the settings to prevent forgery

Syntax: `g_from_noforgeme` bool

g_from_relay - If not authenticated and g_relay_allow_ip matched then block if not local domain or whitelisted

This one helps prevent a local virus sending out spam. It basically says non authenticated users who can relay due to a `g_relay_allow_ip` rule must send from one of your domains or use smtp authentication or be in a white list. Note this test is performed on the message envelope not the body. We recommend insisting on smtp authentication to reduce your risk of this type of problem.

Syntax: `g_from_relay` bool

g_from_relay_white - White list of domains for g_from_relay setting

This is domains that can be used as a 'from' address for non authenticated users, in addition to local domains

Syntax: `g_from_relay_white` string

g_from_rewrite - Rewrite from envelope for outgoing email, e.g. `*@this.domain -> %1@another.domain`

This lets you change the 'from' address from an internal domain name to a valid public domain name. The change is performed on the From envelope (return path), not the from header. And the change does not affect the return path written in local deliveries, only outgoing email. Mfilter rules can be used to rewrite the actual message headers.

Syntax: `g_from_rewrite` was=string to=string

g_from_rewrite_header - Rewrite the from header as well

Replaces the From: header in the message with the new address.

Syntax: `g_from_rewrite_header` bool

g_from_stamp - Stamp if from is probably faked

Stamp message with "X-Verify-Failed:" header if from address is probably faked.

eg: X-Verify-Failed: <user@mydomain.com> From mydomain.com is local but user not authenticated or from g_relay_allow_ip

This check is activated based on the same conditions as g_from_bounce.

Syntax: g_from_stamp bool

g_from_timeout - Timeout on g_badfrom_* checks

Timeout in seconds of g_badfrom_* checks. Default = 60 seconds. If this timeout is reached the g_badfrom check will be classed as having failed.

Syntax: g_from_timeout int

g_from_valid - Require an @ and dotted domain in all return addresses

This forces the sender to either give 'no' reply address or a valid one with an @ and a dotted domain

Syntax: g_from_valid bool

g_gateway - Gateway messages to a particular domain (Or smarthost)

Used to gateway messages to another local mail server. Typically this other server is inside a fire wall so it's local IP address is not known by the DNS server. You specify the domain and IP address to send messages to and this server is treated as 'local' rather than remote in terms of open relay restrictions. eg: nonauthenticated users are able to send in mail. Open relay restrictions do not apply to messages sent to this domain because they are considered as if they were local users and not 'relaying'.

This setting has the fields domain(required), to(required), user(optional), pass(optional), relay=true/false(optional),check=true/false (optional)

Normally "domain" and "to" are the only fields that need to be filled in. eg. To relay mail from anyone to user accounts in the domain somedomain.com to the host 1.2.3.4.

g_gateway domain="somedomain.com" to="1.2.3.4"

user="username" pass="password"

If SMTP authentication is required on the destination server the user and pass fields need to be completed.

check=true

The check=true setting tells surgemail to actually connect to the server and check that recipients exist before accepting an incoming email for that user, this is STRONGLY recommended, as it stops the server having to bounce thousands of messages when spammers send to invalid addresses on your server. If SurgeMail cannot connect it will assume the user does exist so nothing is bounced except when the connection is successful.

Classic smarthost setting

This is where you want to send all outgoing email to another server, that may require authentication, note that we don't use relay="true" as that would make the server an open relay.

g_gateway domain="*" to="isp.mail.server" user="user@isp.server" pass="xxx"

relay="true" (warning, usually not needed or wise, this can make your server into an open relay for spammers to abuse!)

As a safety measure to prevent accidental openrelays, SurgeMail will not relay for non authenticated users or trusted users (users that are allowed to relay due to relaying settings eg g_relay_allow_ip) if the domain is "*". This can be overridden by placing "true" in the "relay" field. eg: To relay all mail for all users to host 1.2.3.4:

g_gateway domain="*" to="1.2.3.4" relay="false"

It is possible to use domain="c:\domains.txt" where domains.txt is a file listing the domains to be gatewayed, this should only be done for one gateway rule, and is only worth doing if you have thousands of domains to gateway.

local="true"

Requires that the destination addresses exist in the local account database.

Gateway after user lookup

When gatewaying to a domain which accepts all email regardless of address (e.g. exchange) you are best to define the users in your local user database, this is the only way to prevent nasty bounces and get rid of all the spam cleanly.

- 1) remove the gateway setting for the domain
- 2) add a virtual domain
- 3) In the virtual domain add surgewall settings, e.g. in this example I'm gatewaying the domain 'netwin.co.nz' to a backend server called 'backend.netwin.co.nz'

```
vdomain address="" name="netwin.co.nz"
...
surgewall "backend.netwin.co.nz"
surgewall_options strip_domain="" proxy_failover="" auth_local="TRUE" pop="" smtp="" imap="" usercgi=""
```

You can find more gateway examples in our FAQ here <http://www.netwinsite.com/surgemail/help/faq.htm#gateway>

Syntax: g_gateway domain=string to=string user=string pass=string relay=string check=bool sms=bool local=bool

g_gateway_allow - Known hosts that act as incoming SMTP or surgewall servers for us
Some spam prevention mechanisms which use the ip address of the incoming system must be disabled for incoming SMTP servers/surgewall/firewall boxes so that stupid limits don't block all the incoming messages from your backup mx server etc. Settings this affects: g_tarpit_max, g_tarpit_max_remote, g_con_perip, RBL checks,

Syntax: g_gateway_allow string

See also: [g_smtp_auth_debug](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#)

g_gateway_always - Always send to gateway even if local domain exists

Always send to gateway even if local domain exists. Not sure why you would want to use this setting other than to temporarily send mail on to another server whilst keeping the local domain and accounts intact and untouced.

Syntax: g_gateway_always bool

g_gateway_auth - Send SMTP auth requests to another host

Send SMTP auth requests to another host.

Syntax: g_gateway_auth string

g_gateway_data - Gateway at the data stage

To allow bounces to be handled cleanly gateway messages before responding to the data command so bounces can go direct without being generated and creating back scatter.

Syntax: g_gateway_data bool

g_gateway_from - Pass 'from' header thru during gateway check

In some cases to verify an email address the correct 'from' must be passed through, normally this is a bad idea as it will cause spf failures, but it is sometimes necessary

Syntax: g_gateway_from bool

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp auth off](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surgeweb backend smtp](#), [surgeplus smtp server name](#), [g_disable smtp after](#), [g_dbabble smtp port](#), [g_dbabble smtp prefix](#), [g_deny smtp](#), [g_safe smtp](#), [g_manager smtp](#), [g_smtp_auth debug](#), [g_smtp bounce nslow](#), [g_smtp cmd timeout](#), [g_smtp data timeout](#), [g_smtp delay stamp](#), [g_smtp welcome delay](#), [g_smtp log protocol](#), [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#),

[g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_help_disable](#), [g_smtp_cram_enable](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#), [g_smtp_thread](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauth](#), [g_smtp_noauthm](#), [g_smtp_noauth_msg](#), [g_verify_smtp](#), [g_surgeplus_smtp_server_name](#)

g_gateway_helo - Header that must exist in incoming bounces (g_send_helo) or bounces are dropped
An incoming filter can discard the majority of incoming bounces by using this setting to figure out if a bounce is valid without having to do a user lookup first! Usually this would be the setting g_send_helo from your 'outgoing' mail server, this setting can be a list of host names.

Syntax: g_gateway_helo string

See also: [send_helo](#), [g_ban_helo](#), [g_helo_optional](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_verify_helo](#)

g_gateway_ifnot - Send mail to gateway in preference to local delivery unless IP matches

The use of g_gateway_ifnot will deliver mail to the g_gateway rule in preference to local delivery unless the IP number matches. This would typically be used to pass mail through an external SMTP server for certain or all domains for scanning purposes etc.

Syntax: g_gateway_ifnot string

g_gateway_ignorewild_ip - Ignore * gateway rules if from ip matches (allows outbound email scanning using gateway * to external scanner)

This setting has no further documentation currently available

Syntax: g_gateway_ignorewild_ip string

g_gateway_mx - If specified IP address is found in mx record for destination then allow relay (not recommended)

This can be useful if you have thousands of servers using your machine for mx backup and you want to allow them simply because the mx records exist, it's much better to use g_gateway or g_relay settings instead as this saves lookups and makes the results entirely more predictable :-)

Syntax: g_gateway_mx string

See also: [disable_smtp_after](#), [old_smtphost](#), [old_smtphost_skip](#), [smtp_auth_off](#), [smtp_welcome](#), [smtp_welcome_name](#), [smtp_from_ip](#), [surgeplus_backend_smtp](#), [surgeplus_smtp_server_name](#), [g_disable_smtp_after](#), [g_dbabble_smtp_port](#), [g_dbabble_smtp_prefix](#), [g_deny_smtp](#), [g_safe_smtp](#), [g_manager_smtp](#), [g_smtp_auth_debug](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_help_disable](#), [g_smtp_cram_enable](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#), [g_smtp_thread](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauth](#), [g_smtp_noauthm](#), [g_smtp_noauth_msg](#), [g_verify_smtp](#), [g_surgeplus_smtp_server_name](#)

g_gateway_orcpt - Writes an original receipt header when forwarding a message, this may disclose multiple recipients, cc/bcc etc use only for tracking faults

This writes a header X-Rcpt-Original: ..., when forwarding a message to another server, good for tracking problems. This may disclose multiple hidden recipients, it should not be used normally

Syntax: g_gateway_orcpt bool

g_group_field - Group Field from authentication database

Based upon a match on an arbitrary field in the authentication database a user can be defined as being part of an access_group. All fields (field, value, group) are required. eg: To add the user to the access_group "paid_user" if the field "mystatus" has the value "fullaccess":

g_group_field field="mystatus" value="fullaccess" group="paid_user"

Syntax: g_group_field field=string value=string group=string

g_hack_detect_disable - Stop admin emails when users login with a weak password

Useful if you must have weak passwords for some reason

Syntax: g_hack_detect_disable bool

g_hacker_max - Login guesses for one ip address before we lockout the ip address

Stops hackers from guessing passwords every day until they find one

Syntax: g_hacker_max int

g_hacker_poison - Poison accounts that instantly blacklist ip address

If user tries to login with this account then their ip address is blocked from further logins

Syntax: g_hacker_poison string

g_hacker_whitelist - Ip addresses to avoid guessing issues

Whitelist for gateways or other systems that you expect multiple failed logins from (e.g. webmail host)

Syntax: g_hacker_whitelist string

g_header_out - Header to add to outgoing posts

Mail header to add to outgoing mailing list posts.

Syntax: g_header_out string

g_header_strip - Strip listed headers from incoming messages

Useful for stripping headers that you don't trust or don't want for some reason

Syntax: g_header_strip string

g_helo_optional - Make the SMTP Helo optional

Helo is optional for SMTP protocol (not recommended).

Syntax: g_helo_optional bool

g_home - Root directory of the mail server

This setting controls where the mail server runs including the many sub directories it creates below this directory for work files and log files for each domain. Not something you should generally change.

Syntax: g_home string

g_honeypot_key - Key for HTTP RBL service www.projecthoneypot.org - not recommended

Do not share your key you can get a key for free from this web site. By defining this setting you will enable honeypot lookups, which in turn will block web imap pop and smtp authentication connections from listed sites, it does not block normal incoming email, but does reduce the permitted guess count to '1'. You can whitelist an ip address using g_spam_allow or g_hacker_whitelist, this setting will tend to cause false positives which will stop users logging in, we don't recommend you use this setting currently.

Syntax: g_honeypot_key string

g_honeypot_rbl - RBL name to lookup, typically dnsbl.httpbl.org

This is the name of the rbl database we are going to query

Syntax: g_honeypot_rbl string

g_http_proxy - Proxy web server for fetching files via HTTP

Proxy web server for fetching files if direct access fails. (mainly for updates to the spam prevention rules from netwinsite.com and for downloading the latest version of the SurgePlus Windows client to make available to your users.)

Syntax: g_http_proxy string

g_imap_acl - Enable ACL (shared folders) in imap

This setting allows folders to be shared between users. See the domain setting 'imap_public'. Requires surgemail 3.9d or later! For this to work you will need an imap client that supports ACL's to create and map shared folders (.e.g. thunderbird)

Syntax: g_imap_acl bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_if](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_blacklist - Test if imap users are in rbl's and email admin

This lets you find any of your users who's ip address has been blacklisted, at most it will email once a day, any additional entries are logged in mail.err log file (search for 'blacklist')

Syntax: g_imap_blacklist bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_if](#), [g_imap_acl](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_capa - Where to get the CAPABILITY value from

When you have suffix based domains and you're using SurgeWall the CAPABILITY request comes before the domain of the user is known. As such SurgeMail cannot determine whether to send the real servers CAPABILITY or it's own. This setting will choose the default behaviour, valid values are: Local, . By default SurgeMail defaults to the behaviour of the primary domain, if it's surgewall then it obtains the real server capability. "Local" defaults to SurgeMails own capability, and defaults to the real server capability.

Syntax: g_imap_capa string

g_imap_capa_strip - Capability values to hide

In some situations you might not want to advertise server capabilities, for example SURGEMAIL and XFLDDATA when they cause problems with SurgeWall operations. Or perhaps the IDLE capability. Specifying the capability strings to hide here will cause SurgeMail to stop advertising those capabilities.

Syntax: g_imap_capa_strip string

g_imap_cram_enable - Enable CRAM-MD5 authentication (requires nauth 4.0h or greater)
Please note that CRAM-MD5 does have security implications, specifically it means that the local users password must be stored in a semi reversable state in the authent database. Also you must be using the new version of the NWAauth module.

Syntax: g_imap_cram_enable bool

See also: [smtp_auth_off](#), [surgeauth_auth](#), [g_acctlog_authonly](#), [g_allow_user_authent_field_get](#), [g_allow_user_authent_field_set](#), [g_authent_always](#), [g_authent_any](#), [g_authent_allow_badascii](#), [g_authent_prefix_sep](#), [g_authent_process](#), [g_authent_cachelife](#), [g_authent_cachebad](#), [g_authent_cachesize](#), [g_authent_domain](#), [g_authent_encrypt_key](#), [g_authent_number](#), [g_authent_info](#), [g_authent_info_grp](#), [g_authent_ip](#), [g_authent_path_broken](#), [g_authent_single](#), [g_authent_spaces](#), [g_authent_strip_domain](#), [g_authent_restart](#), [g_authent_logall](#), [g_authent_fwdfile](#), [g_authent_timeout](#), [g_authent_last_login](#), [g_auth_hide](#), [g_auth_norelay](#), [g_auth_skipgateway](#), [g_mirror_nauth](#), [g_mirror_nauth_always](#), [g_filter_pipe_nauth](#), [g_gateway_auth](#), [g_smime_skip_auth](#), [g_smtp_auth_debug](#), [g_smtp_portauth](#), [g_smtp_etrn_auth](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_nauth](#), [g_smtp_nauthm](#), [g_smtp_nauth_msg](#), [g_spam_check_auth](#), [g_xauthuser_hide](#)

g_imap_delay - Glob data into bigger packets, never use this

This setting has no further documentation currently available

Syntax: g_imap_delay bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_friends - Make the friends_pendign folder visible in imap

Setting to map the friends_pending folder into an imap folder. There is no corresponding setting for the 'held' folder as we believe people should always use the friends mechanism as it is a superset of the held folder in functionality

Syntax: g_imap_friends bool

g_imap_idle_nsf - The number of seconds before a complete directory rescan. To be use on NSF network drives

Number of seconds for IMAP IDLE to do directory rescan

Syntax: g_imap_idle_nsf int

g_imap_log_body - Log imap fetch body commands to msg*.rec log files

This only logs when a body or body part is read via imap

Syntax: g_imap_log_body bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#),

[g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_log_copy - Log imap copy commands to msg*.rec log files

This setting has no further documentation currently available

Syntax: `g_imap_log_copy bool`

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_log_flush - IMAP log flush

Flush IMAP log on every write (for debugging).

Syntax: `g_imap_log_flush bool`

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_log_header - Log imap fetch header commands to msg*.rec log files (not usually needed)

This logs rather a lot so may create excessive logging. Probably the log body setting is more wise.

Syntax: `g_imap_log_header bool`

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_log_protocol - Log IMAP protocol

Log IMAP protocol and other IMAP information to the mail.log file.

Syntax: `g_imap_log_protocol bool`

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_max_messages - The number of messages in a single imap folder, default 200000

This setting helps limit impact when a user has a large folder, it will fail to load a folder larger than this and report errors in the log, it does not prevent the folder from having messages added to it, and it does not inform the user that the problem has occurred, this setting is primarily to limit impact of a crazy user :-)

Syntax: g_imap_max_messages int

g_imap_no_internal_date - Disable the internal date output on IMAP commands

The RFC implementation of internal date is broken with MS outlook. SurgeMail has been modified to conform to the outlook implementation of internal date making this setting redundant..

Syntax: g_imap_no_internal_date bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_old - Revert to old imap module

Replace normal imap with old imap module, not recommended/supported

Syntax: g_imap_old bool

g_imap_old_ip - Revert to old imap module for some ip's

Replace normal imap with old imap module, not recommended/supported

Syntax: g_imap_old_ip string

g_imap_pop_burst - Always burst using imap code

Prevents re downloading messages if file indicating user is using imap is lost. Generally this setting is not needed and should not be used. Turning it on/off will result in users getting duplicate messages if they are using POP and have leave on server ticked

Syntax: g_imap_pop_burst bool

g_imap_port - IMAP Port (default 143)

Specifies the PORT to listen for IMAP connections on. IMAP is an alternative to POP protocol where the messages and folders all exist on the server. This is ideal when sharing a mail account between several users or when using Email from more than one computer. Use the keyword 'disabled' to disable this part of the surgemail service.

Syntax: g_imap_port int

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_search_noattach - Skip non text attachments when searching

This setting has no further documentation currently available

Syntax: g_imap_search_noattach bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_secure_port - IMAP Port (default 993)

Specifies the PORT to listen for dedicated SSL IMAP connections.

Syntax: g_imap_secure_port int

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_size_fetch - If true, will display message sizes on fetch command. (ie * 123 EXISTS)

Displays message size in IMAP responses

Syntax: g_imap_size_fetch bool

g_imap_spam_train - Train if moving message to 'spam' folder, or from 'spam' folder to inbox

This setting has no further documentation currently available

Syntax: g_imap_spam_train bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_testing - Test imap module instead of normal one (not functional)

Replace normal imap with a test one, this is not functional, do not use this setting.

Syntax: g_imap_testing bool

g_imap_timeout - Time, in minutes for imap timeout, RFC required default is 30

You may in some cases wish to reduce this below the RFC required default if your server is under very heavy load. Results may be unexpected when breaking RFC behavior!

Syntax: g_imap_timeout int

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#),

[g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_timezone - Timezone to display - for testing purposes only

as per title :-)

Syntax: g_imap_timezone string

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_lf](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

g_imap_uidl_nofix - Disable UIDL auto repair of duplicate entries

If true disable auto repair of identical UIDL entries.

Syntax: g_imap_uidl_nofix bool

g_imap_user_flags - This setting may confuse some email clients (mac) use with caution
This may confused some email clients if multiple clients are used on a single account as the user flags can conflict

Syntax: g_imap_user_flags bool

g_include - Include another ini file global settings only

Unlike the include command this setting will allow editing of the ini file in web admin, but settings included via this setting will not appear in the admin interface

Syntax: g_include string

g_iplimit - Untrusted local ip addresses e.g. web servers, special sending limits applied.
These limit settings let you control untrusted sources which may get viruses or cgi scripts that open them up to abuse. By throttling the remote addresses limit this will prevent any significant abuse. Authenticated sessions are 'not' limited!.

Syntax: g_iplimit string

g_iplimit_islocal - Add domains to list of domains considered local for limit counting
See explanation of g_iplimit

Syntax: g_iplimit_islocal string

g_iplimit_local - Max sends from untrusted ip to local domains per 30 minutes.
See explanation of g_iplimit

Syntax: g_iplimit_local int

g_iplimit_remote - Max sends from untrusted ip to remote domains per 30 minutes.
See explanation of g_iplimit

Syntax: g_iplimit_remote int

g_iplimit_whitelist - List of 'from' addresses that should bypass limits
This lets you bypass the iplimit restrictions for a known trusted user/form that needs to send a lot of

local/remote emails

Syntax: `g_iplimit_whitelist` string

`g_ipv6_enable` - Enable IPV6 networking (beta testing only)

Enable IPV6 networking - this feature only exists on some platforms. And is not yet fully implemented. We strongly advise you DO NOT use this on a live mail server but rather use it in your ipv6 test environment.

Syntax: `g_ipv6_enable` bool

See also: [dmail_skip_imap](#), [imap_public](#), [old_imaphost](#), [old_imaphost_always](#), [old_imaphost_createuser_disable](#), [old_imaphost_nodomain](#), [old_imaphost_nodelete](#), [old_imaphost_prefix](#), [old_imaphost_file](#), [old_imaphost_user](#), [old_imaphost_pass](#), [old_imaphost_lowercase](#), [old_imaphost_skip](#), [g_autologin_imap_disable](#), [g_fix_imap_if](#), [g_imap_acl](#), [g_imap_blacklist](#), [g_imap_cram_enable](#), [g_imap_capa](#), [g_imap_capa_strip](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_imap_port](#), [g_imap_delay](#), [g_imap_secure_port](#), [g_imap_search_noattach](#), [g_imap_spam_train](#), [g_imap_no_internal_date](#), [g_imap_timezone](#), [g_imap_timeout](#), [g_imap_uidl_nofix](#), [g_imap_size_fetch](#), [g_imap_idle_nsf](#), [g_imap_testing](#), [g_imap_old](#), [g_imap_old_ip](#), [g_imap_pop_burst](#), [g_imap_friends](#), [g_imap_user_flags](#), [g_imap_max_messages](#), [g_safe_imap](#), [g_old_imap_headbody](#), [g_ssl_allow_imap](#), [g_ssl_require_imap](#)

`g_keepalive` - Attempts to use keepalive for the web sessions (experimental & faulty currently)

Don't use this yet, we are still working on it.

Syntax: `g_keepalive` bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

`g_key_manual` - Try and activate automatically when the key expires

When you purchase updates you must activate to get the expire date reset in surgemail, if this setting is not turned on then surgemail will try and do this automatically for you.

Syntax: `g_key_manual` bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

`g_key_nowarning` - Disable reminders to update your license

Disables the email reminding you to pay for updates for virus and spam filter and new versions etc...

Syntax: `g_key_nowarning` bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

`g_language_default` - Default language for user web interface

If the user has not yet selected a language then this language is used as a default. If the language specified here does not exist in the language files, or nothing is specified here then English is used as the default language.

Syntax: `g_language_default` string

`g_last_login` - Create `last_login.time` files

If true then when users login via pop or imap or webmail the file `last_login.time` is created/touched, this can then be used by local scripts to determine which user directories are not in active use.

Syntax: `g_last_login` bool

`g_last_login_days` - If last login is more than this many days then reject email - do not use on mirrors
This can be used on a shared disk cluster to establish which users are inactive. On a normal mirror or stand alone system you should use `DISABLE_SMTP_AFTER`

Syntax: `g_last_login_days` int

`g_late_forward` - Apply all users forwarding rules after friends, spam, and filtering

By default users forwarding rules are applied before friends, spam and user filter rules. By default users can tick an option on their forwarding page to perform 'late' forwarding, that is forwarding that occurs after friends, spam and filtering. This option overrides the user option and causes all user forwarding rules to be applied after friends, spam and filtering.

Syntax: `g_late_forward` bool

`g_ldap_forward` - Remote ldap server to forward requests to (only for testing do not use)
Forwards all ldap requests to another host, primarily intended for testing, use at your own risk.

Syntax: `g_ldap_forward` string

See also: [ldap_disable](#), [ldap_anydomain](#), [g_ldap_port](#), [g_ldap_outlook_browse_max](#)

`g_ldap_outlook_browse_max` - Basic outlook ldap address browsing, max items (KEEP THIS SMALL eg <50): default=0 (disabled)
numeric maximum items to return default=0 (ie disabled)

Syntax: `g_ldap_outlook_browse_max` int

See also: [ldap_disable](#), [ldap_anydomain](#), [g_ldap_port](#), [g_ldap_forward](#)

`g_ldap_port` - LDAP Port (normally 389)

If specified this enables the mini ldap server inside surgemail which allows users with email clients that can do 'ldap' directory lookups to search for other users on the system. Obviously this should NEVER BE turned on for a public mail server, it is only appropriate with private mail servers where all users who can access the system are trusted.

There are additional 'domain' settings `ldap_anydomain`, which lets users search for users outside their own domain name. And `ldap_disable` which can disable ldap for specific domains.

Syntax: `g_ldap_port` int

See also: [ldap_disable](#), [ldap_anydomain](#), [g_ldap_forward](#), [g_ldap_outlook_browse_max](#)

`g_if_fix_off` - If input contains naked 'lf' characters then reject with error instead of stripping as usual

This setting has no further documentation currently available

Syntax: `g_if_fix_off` bool

`g_local_skipgateway` - Skip gateway rule for local messages

If true skip gateway rule for local messages (bounces etc).

Syntax: `g_local_skipgateway` bool

`g_log_bounce_disable` - Stop bounce reject entries filling up log (typically from spam bounces)
Disables useless logging in `msg*.rec` files, only recommended for busy servers

Syntax: `g_log_bounce_disable` bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_nrotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

`g_log_date` - Log full date in log files
Makes log lines more complete

Syntax: `g_log_date` bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_nrotate](#), [g_log_user](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

`g_log_date_msg` - Log full date in msg log files (`g_log_date` required too)
Makes log lines more complete with the full date

Syntax: `g_log_date_msg` bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#),

[g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_disable - Disable most logging - not recommended

This setting has no further documentation currently available

Syntax: `g_log_disable bool`

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_dns - Log dns responses in gory detail

Useful when debugging unexpected DNS results, search for 'dns' in mail.log to find the results.

Syntax: `g_log_dns bool`

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_log_dropped_disable - Don't log if no 'data' command sent

Disables useless logging in msg*.rec files, only recommended for busy servers

Syntax: `g_log_dropped_disable bool`

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#)

[g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_flush - Flushing log - flush on every write

This makes the server flush log data after every write to the file. This affects performance but can sometimes be the only way to track down an unusual fault eg: if the server dies the log is completely up to date and shows the last thing the server did before dying.

Syntax: `g_log_flush bool`

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_fwd - Log fwd/redirection rules associated in msg.rec

Log fwd/redirection rules associated with `g_log_rcpt` in `msg.rec` files.

Syntax: `g_log_fwd bool`

g_log_level - Set logging level

Set the logging level. This is primarily intended for finding faults with the server. Info level logging is the default. Alternatives are 'error' and 'debug'

Syntax: `g_log_level string`

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#),

[g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_norcpt - Don't log individual recipients in msg.rec files

Log individual recipients in msg.rec files

Syntax: g_log_norcpt bool

g_log_path - Path for log files

Sets the path for all SurgeMails generated logfiles. (except the delivery record logs)

Syntax: g_log_path string

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_pid - Log pid

Log PID along with thread-id in the UNIXlog files.

Syntax: g_log_pid bool

g_log_reject_disable - Disable the logging of rejected mail

SurgeMail will normally log failed deliveries due to MFilter / SmiteSpam / etc in the delivery logs. This setting will restrict this logging to accepted mail only.

Syntax: g_log_reject_disable bool

g_log_size - Size of the mail.log files before they are rotated

The mail.log files are a fixed size rotating log of what is happening inside SurgeMail. Dependant on the load of your server this may contain a few days worth of activity or a few minutes worth. This setting allows you to change the default 2MB before rotation size.

Syntax: g_log_size int

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#),

[g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_slow - Do slower logging system

Forces logging to disk even if it may slow things down. Not recommended.

Syntax: g_log_slow bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_start_norotate - Don't rotate log on startup

By default the mail.log is rotated to mail2.log... on startup.

Syntax: g_log_start_norotate bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_log_syslog - Send 'msg.rec' entries to syslog

This is useful to 'merge' log information on a single host, on unix you specify the destination in your syslog configuration rather than specifying a host. On windows you can specify the remote host as you may not have a local syslog daemon

Syntax: `g_log_syslog` bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

`g_log_syslog_debug` - Send 'mail.log' entries to syslog as 'mail.debug' data

This data is probably not worth sending to syslog, it's really debugging information of no long term value and too much to store.

Syntax: `g_log_syslog_debug` bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

`g_log_syslog_host` - Specify host to send syslog entries to (windows only)

On windows this lets you tell surgemail where the syslog daemon is, on unix you can do this in your syslog config file.

Syntax: `g_log_syslog_host` string

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#),

[g blogs maximum image size](#), [g blogs maximum items in top page](#), [g blogs max per user](#),
[g blogs default template](#), [g blogs use sub domains](#), [g blogs sub domain prefix](#), [g blogs not unique](#),
[g blogs not global](#), [g blogs no suffix](#), [g blogs ping](#), [g blogs domonly](#), [g blogs image optional](#),
[g blogs allow links](#), [g blogs cleanup links](#), [g blogs comment rev](#), [g imap log protocol](#),
[g imap log flush](#), [g imap log copy](#), [g imap log header](#), [g imap log body](#), [g last login](#),
[g last login days](#), [g log flush](#), [g log fwd](#), [g log level](#), [g log disable](#), [g log path](#), [g log pid](#), [g log thid](#),
[g log reject disable](#), [g log bounce disable](#), [g log dropped disable](#), [g log tcp read](#), [g log tcp write](#),
[g log norcpt](#), [g log size](#), [g log dns](#), [g log slow](#), [g log start norotate](#), [g log user](#), [g log date](#),
[g log date msg](#), [g log syslog](#), [g log syslog debug](#), [g log syslog only](#), [g msg log extra](#),
[g setpassword firstlogin](#), [g perflg disable](#), [g perflg flush interval](#), [g perflg lowres](#), [g perflg logall](#),
[g perflg surgeonly](#), [g smtp log protocol](#), [g smtp log size](#), [g spawn log](#), [g spf debug log](#),
[g spflog enable](#), [g spflog domains](#), [g ssl require login](#), [g surgeblog](#), [g user blogs](#),
[g user friends domain log disable](#), [g user friends log disable](#), [g surgeweb logall](#),
[g surgeplus log level](#)

g_log_syslog_only - Disable writing to msg.rec

This prevents the local logs from being written

Syntax: g_log_syslog_only bool

See also: [blogs_max_per_user](#), [loginfails](#), [url blogs](#), [g acctlog sum inactive](#), [g acctlog authonly](#),
[g acctlog noaliases](#), [g alias login disable](#), [g bad login mins](#), [g bad login allow](#), [g bad login ip allow](#),
[g bad login ip ignore](#), [g bank log](#), [g authent logall](#), [g authent last login](#), [g autologin pop](#),
[g autologin file](#), [g autologin imap disable](#), [g blogs enable](#), [g blogs maximum image width](#),
[g blogs maximum image size](#), [g blogs maximum items in top page](#), [g blogs max per user](#),
[g blogs default template](#), [g blogs use sub domains](#), [g blogs sub domain prefix](#), [g blogs not unique](#),
[g blogs not global](#), [g blogs no suffix](#), [g blogs ping](#), [g blogs domonly](#), [g blogs image optional](#),
[g blogs allow links](#), [g blogs cleanup links](#), [g blogs comment rev](#), [g imap log protocol](#),
[g imap log flush](#), [g imap log copy](#), [g imap log header](#), [g imap log body](#), [g last login](#),
[g last login days](#), [g log flush](#), [g log fwd](#), [g log level](#), [g log disable](#), [g log path](#), [g log pid](#), [g log thid](#),
[g log reject disable](#), [g log bounce disable](#), [g log dropped disable](#), [g log tcp read](#), [g log tcp write](#),
[g log norcpt](#), [g log size](#), [g log dns](#), [g log slow](#), [g log start norotate](#), [g log user](#), [g log date](#),
[g log date msg](#), [g log syslog](#), [g log syslog debug](#), [g log syslog host](#), [g msg log extra](#),
[g setpassword firstlogin](#), [g perflg disable](#), [g perflg flush interval](#), [g perflg lowres](#), [g perflg logall](#),
[g perflg surgeonly](#), [g smtp log protocol](#), [g smtp log size](#), [g spawn log](#), [g spf debug log](#),
[g spflog enable](#), [g spflog domains](#), [g ssl require login](#), [g surgeblog](#), [g user blogs](#),
[g user friends domain log disable](#), [g user friends log disable](#), [g surgeweb logall](#),
[g surgeplus log level](#)

Syntax: g_log_tcp_read string

See also: [blogs_max_per_user](#), [loginfails](#), [url blogs](#), [g acctlog sum inactive](#), [g acctlog authonly](#),
[g acctlog noaliases](#), [g alias login disable](#), [g bad login mins](#), [g bad login allow](#), [g bad login ip allow](#),
[g bad login ip ignore](#), [g bank log](#), [g authent logall](#), [g authent last login](#), [g autologin pop](#),
[g autologin file](#), [g autologin imap disable](#), [g blogs enable](#), [g blogs maximum image width](#),
[g blogs maximum image size](#), [g blogs maximum items in top page](#), [g blogs max per user](#),
[g blogs default template](#), [g blogs use sub domains](#), [g blogs sub domain prefix](#), [g blogs not unique](#),
[g blogs not global](#), [g blogs no suffix](#), [g blogs ping](#), [g blogs domonly](#), [g blogs image optional](#),
[g blogs allow links](#), [g blogs cleanup links](#), [g blogs comment rev](#), [g imap log protocol](#),
[g imap log flush](#), [g imap log copy](#), [g imap log header](#), [g imap log body](#), [g last login](#),
[g last login days](#), [g log flush](#), [g log fwd](#), [g log level](#), [g log disable](#), [g log path](#), [g log pid](#), [g log thid](#),
[g log reject disable](#), [g log bounce disable](#), [g log dropped disable](#), [g log tcp write](#), [g log norcpt](#),
[g log size](#), [g log dns](#), [g log slow](#), [g log start norotate](#), [g log user](#), [g log date](#), [g log date msg](#),
[g log syslog](#), [g log syslog debug](#), [g log syslog only](#), [g log syslog host](#), [g msg log extra](#),
[g setpassword firstlogin](#), [g perflg disable](#), [g perflg flush interval](#), [g perflg lowres](#), [g perflg logall](#),

[g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#),
[g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#),
[g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#),
[g_surgeplus_log_level](#)

g_log_tcp_read,g_log_tcp_write - Log actual data for a specific IP

These settings let you 'trace' the data going 'to' and or 'from' a specific IP address (or list, or wild card) Lets say you have a client on a specific address that has a problem where the fault could be server/client or network related. To track it down add this to surgemail.ini

```
g_log_tcp_read "2.3.4.5"
g_log_tcp_write "2.3.4.5"
```

Then try whatever is 'failing' and examine 'mail.log' to see what was read/written to that client.

Syntax: g_log_tcp_write string

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#),
[g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#),
[g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#),
[g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#),
[g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#),
[g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#),
[g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#),
[g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#),
[g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#),
[g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#),
[g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#),
[g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#),
[g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#),
[g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#),
[g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#),
[g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#),
[g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#),
[g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#),
[g_surgeplus_log_level](#)

g_log_thid - Log thread id in .rec files

Logs the thread id in the msg*.rec files, this is good for some types of debugging.

Syntax: g_log_thid bool

See also: [redirect](#), [redirect_max](#), [redirect_cc](#), [redirect_hash](#), [g_create_record_ip](#), [g_bounce_redirect](#),
[g_orbs_rec](#), [g_received_name](#), [g_received_names](#), [g_received_skip](#), [g_received_skip_all](#),
[g_received_skip_spf](#), [g_recent_bypass](#), [g_record_days](#), [g_record_hash](#), [g_record_path](#), [g_redirect](#),
[g_redirect_cc](#), [g_redirect_from](#), [g_redirect_from_cc](#), [g_redirect_hide](#), [g_redirect_iflocal](#),
[g_redirect_ignore_errors](#), [g_redirect_noautocreate_rules](#), [g_spam_allow_recent](#), [g_user_receive_rule](#),
[g_virus_recent_skip](#)

g_log_user - Log pop/imap/smtp protocol for specified user

Creates a file for each user that matches this list, user_user@domain.log

Syntax: g_log_user string

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#),
[g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#),
[g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#),
[g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#),
[g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#),
[g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#),
[g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#),
[g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#),
[g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#),
[g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#),
[g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#),

[g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_lookup_names - Lookup names for connecting IP addresses

This is one of those things that you very likely do not want to turn on. It makes the mail server lookup the IP name of any connecting user, however lookups can take 30-90 seconds so it can negatively impact apparent performance. Most of the access rules in the server can accept IP names if this setting is enabled, e.g. instead of specifying local users are 153.2.3.* you can say '*.netwinsite.com'

Syntax: `g_lookup_names bool`

g_lookup_reject_fails - If lookup cannot get a name, reject user (not generally recommended)

If lookup cannot get a name, reject user (not generally recommended)

Syntax: `g_lookup_reject_fails bool`

g_lowdisk_warning - Disk space level below which to warn the manager

SurgeMail checks available disk space on startup and every half hour whilst running on all the mail, temp and home directories. If any is found to be low an email is sent to the system manager. The recommended level is at least 100MB (default is 10MB).

Syntax: `g_lowdisk_warning string`

g_mailbox_path - Default directory to store mail

Default directory to store mail this is used to set mailbox_path when creating domains.

Syntax: `g_mailbox_path string`

g_maildir_max - Max messages in a folder, do not adjust

The default is 30,000. When exceeded additional messages are invisible until some are deleted. We strongly recommend you don't change this limit as large folders are geometrically inefficient and users should take steps to avoid this limit rather than increasing it.

Syntax: `g_maildir_max int`

See also: [g_maildir_netwin](#), [g_maildir_standard](#), [g_maildir_report](#)

g_maildir_netwin - Use NETWIN proprietry storage format - Not Recommended

This changes the storage format from one message per file, to a proprietry format, the spool is converted automatically when you restart surgmail. As a new feature which reformats all messages stored this settings has some risks, we suggest caution particularly on an existing server, ensure you have a backup mechanism of some kind in place!. Although this setting can give performance gains we think generally the gains do not out weigh the risk introduced, personally I prefer a simple 'directory of files' for each mail folder

Syntax: `g_maildir_netwin bool`

g_maildir_report - Email manager on ndb errors

This is for debugging and not for general use

Syntax: `g_maildir_report bool`

g_maildir_standard - Use more standard maildir format

The maildir format is flawed in that it is not designed to be used on Windows systems. This setting will force SurgeMail to use a more standard maildir format, but does mean you cannot just

copy mail from a UNIX box to a Windows box as the ":" character is a reserved character on Windows systems.

Syntax: g_maildir_standard bool

g_mailstatus_message - Error message to give when mailstatus is set to specified state

This allows you to specify the error message given to the user when they are set to certain states, you may use other authentic fields in the message, for example:

g_mailstatus_message state="payup" message="Payment is due \$full_name\$, please pay here: <http://your.site/path/file.htm>"

Syntax: g_mailstatus_message state=string message=string

g_manager - Email address of manager

Email address to send reports to.

Syntax: g_manager string

g_manager_port - Manager port (default 7026)

This is the port the web manager and web mail access will run on. By default it is port 7026. Use the keyword 'disabled' to disable this part of the surgemail service.

Syntax: g_manager_port int

g_manager_secure_port - Manager secure port (default 143)

This should be the main server management port and provides a secure server management connection. By default it is port 7025. <https://your.mail.server:7025>. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_manager_secure_port int

g_manager_smtp - SMTP server for manager Emails about failures

For obvious reasons, if the server is not working it cannot use itself to send the manager an Email message, so for highest reliability you may want to define another mail server for fault reports to be Emailed to.

Syntax: g_manager_smtp string

g_manager_username - Global domain managers username (for web based domain administration)
Specifies the local users which have manager rights for all domains. These users can login to the user self management interface and will receive special domain manager options. This setting works slightly different to the domain level 'manager_username' setting in that if you specify an account without the @domain part i.e. 'admin' it gives all admin users in all domains domain rights over all domains.

Syntax: g_manager_username string

g_max_bad_ip - Max bad recipients per ip address before blocking that ip

This setting is important to stop hackers fishing for email addresses by guessing, I recommend you start with a low setting like 5, but increase to 100 if it causes problems. If you have a firewall or spam filter in front of surgemail add G_SPAM_ALLOW to whitelist it's ip address

Syntax: g_max_bad_ip int

g_max_bad_ip_time - Seconds to block guessing hackers

The default is 1 day (used to be 1 hour). Units is seconds

Syntax: g_max_bad_ip_time int

g_max_bad_nolookup - Max bad recipients in a row if exceeded skip user lookup

Max bad recipients in a row if exceeded skip user lookup - useful when tarpitting a spammer.

Syntax: g_max_bad_nolookup int

g_max_bad_to - Max bad recipients in a row

If a system sending your system Email sends more than the specified number of bad addresses in a row then it is assumed to be incoming spam and further messages are rejected.

Syntax: `g_max_bad_to` string

g_mdir_hash - SurgeMail hashing mode

Hashing mode for SurgeMail, default is 5, for compatibilty with /b/o/bob use 2.

Syntax: `g_mdir_hash` int

g_mdir_prefix - Maildir folder prefix

Prefix for maildir folders defaults to 'mdir', use '.' for compatibility with qmail.

Syntax: `g_mdir_prefix` string

g_mfilter_addonly - Add headers only

If true then only allow 'adding' headers, not changing them.

Syntax: `g_mfilter_addonly` bool

g_mfilter_bounces - Run mfilter on bounce messages and responders etc Run the mfilter processing even on bounces

Syntax: `g_mfilter_bounces` bool

g_mfilter_file - Path to mfilter.rul spam rule processing

This is the full path to the Mfilter rule file which provides advanced message filtering capabilities. See [Mfilter.htm](#) for more details.

Syntax: `g_mfilter_file` string

g_mfilter_localonly - Only filter local deliveries

If true then only run Mfilter on local deliveries.

Syntax: `g_mfilter_localonly` bool

g_mfilter_maxlen - Mfilter Max message length

Size to truncate messages to before processing with Mfilter.

Syntax: `g_mfilter_maxlen` int

g_mfilter_noisey - Do log anything in mfilter

Logs the real details of mfilter, never user on a live busy system this is only intended for debugging an mfilter script. It logs every line of the script!

Syntax: `g_mfilter_noisey` bool

See also: [g_mfilter_file](#), [g_mfilter_bounces](#), [g_mfilter_maxlen](#), [g_mfilter_addonly](#), [g_mfilter_localonly](#), [g_mfilter_trace](#), [g_mfilter_skip_ip](#), [g_mfilter_skip_from](#), [g_mfilter_skip_to](#), [g_user_mfilter](#)

g_mfilter_skip_from - From addresses (envelope) to skip mfilter processing for

This setting has no further documentation currently available

Syntax: `g_mfilter_skip_from` string

g_mfilter_skip_ip - Skip mfilter for messages from these ip's

This allows you to add a comma separated list of ip's to skip running mfilter on. This is based on the ip of the sender. Wild cards and ranges can be used.

Example:

`g_mfilter_skip "10.0.0.2,210.56.43.*,193.1.16-24.0-255"`

Syntax: `g_mfilter_skip_ip` string

g_mfilter_skip_to - To addresses to skip mfilter processing for

If one matches then mfilter is skipped for entire message

Syntax: g_mfilter_skip_to string

g_mfilter_trace - Log trace lines in Mfilter

Log trace lines in Mfilter for debugging .

Syntax: g_mfilter_trace bool

g_mirror_config - Mirror surgemail.ini

Syntax: g_mirror_config "true/false"

You put this on both machines and it will attempt to mirror the surgemail.ini. There will be some settings that you do not wish to mirror and these can be exempted by using:

[g_mirror_config_except](#) "setting,setting,setting"

Some settings are not mirrored by default these are: g_mirror_host, g_mirror_nwauth*, g_mirror_mode, g_authent_path, g_dlist_path, g_log_path, g_record_path, g_home, g_authent_process, g_mfilter_file, g_webmail_work, g_work, g_virus_cmd, g_atrn_port, g_imap_port, g_imap_secure_port, g_ldap_port, g_manager_port, g_manager_secure_port, g_monitor_port, g_pop_port, g_pop_secure_port, g_ppd_port, g_smtp_port, g_smtp_secure_port, g_webmail_port, g_webmail_secure_port, g_surgeplus_port, g_surgeplus_secure_port, g_surgeplus_web_port, g_bind_out, g_virus_avast, dmail_drop_path, dmail_bin_path, web_path, webmail_work

(it is possible we will update this list over time)

* g_mirror_nwauth is obsolete don't use it.

Syntax: g_mirror_config bool

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune age](#), [g_mirror_threads](#), [g_mirror live](#), [g_mirror live max](#), [g_mirror_config_except](#), [g_mirror trash](#), [g_mirror debug](#)

g_mirror_config_except - Mirror surgemail.ini

Syntax: g_mirror_config "setting,setting,setting"

This will tell the server not to import the specified settings from the other mirror.

Example:

[g_mirror_except](#) "g_spam_allow"

This will tell the server not to change this setting. This only affects the machine its on, if the other server does not have this set, it will continue to mirror the setting. This setting accepts wildcards. This setting accepts a special case value "address" that will prevent mirroring of existing domain ip addresses, allowing different ips on each mirror machine. There are a number of settings which are not mirrored by default these are specified above in [g_mirror_config](#).

In addition the mailbox_path setting is not mirrored, **unless**, the existing setting is a sub directory of the g_mailbox_path and the new setting is a sub directory of the g_mailbox_path from the other server, in which case the mailbox_path is set to the same sub directory using the existing g_mailbox_path setting eg.

```
[recieving server]
g_mailbox_path "c:\surgemail\mbox"
mailbox_path "c:\surgemail\mbox\domain"

[sending server]
g_mailbox_path "d:\surgemail\mbox"
mailbox_path "c:\surgemail\mbox\domain_moved_here"

[result on recieving server]
g_mailbox_path "c:\surgemail\mbox"
mailbox_path "c:\surgemail\mbox\domain_moved_here"
```

Syntax: g_mirror_config_except string

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune age](#), [g_mirror_threads](#), [g_mirror live](#), [g_mirror live max](#), [g_mirror_config](#), [g_mirror trash](#), [g_mirror debug](#)

g_mirror_debug - Log more info to mirror log.

Helps when tracking down fault with nwauth or mirroring, never leave turned on as it can lead to mutex crashing

Syntax: g_mirror_debug bool

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#)

g_mirror_host - Mirror host

This unique SurgeMail feature allows you to setup two identical mail servers across a local or widearea network. The waiting mail messages & folders etc are duplicated continuously between the two systems, so users can use either system. If either system fails for any hardware reason the other acts as an instant on line replacement without any interruption to the user. In addition when the faulty system is replaced the two automatically re-synchronize.

[See this page for Mirror overview](#)

Syntax: `g_mirror_host string`

See also: [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_live - Mirror: Send incoming messages immediately

Enables a faster mirroring mechanism, strongly recommended, this setting will be the default in a future release

Syntax: `g_mirror_live bool`

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_live_max - Limit size of mirror_live default 60k

This prevents smtp delays when mirroring over a slowish link. The default is 60k

Syntax: `g_mirror_live_max int`

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_mode - Master / slave mirror system

Certain actions may only be run on the mirror master system (such as expire processing) or are different in behaviour between the master and slave (such as NWAuth mirroring and dlist mirroring). This setting must be set to MASTER on one system and SLAVE on the other system for correct operation. (Note basic mirroring of delivered mail will happen if this setting is the same on both systems it is just some of the special mirroring functionality that this is required for)

Syntax: `g_mirror_mode string`

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_nossl - Disable SSL for mirror protocol connection

This is best turned off unless your servers are talking over a wide area untrusted network.

Syntax: `g_mirror_nossl bool`

See also: [g_mirror_host](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_nwauth_always - Mirror nwauth database files

Set this if you're using multiauth to run nwauth and you want those files mirrored. Requires you to add -isslave2 to multiauth.ini nwauth command line. Requires the nwauth files to be located in the surgemail root/install directory.

Syntax: `g_mirror_nwauth_always bool`

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_prune_age - Mirror minimum age for items to be pruned during sync_prune

Mirror minimum age for items to be pruned during sync_prune, default 14 days.

Syntax: g_mirror_prune_age int

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_secret - Mirror secret shared password

This password is required to prevent the mirroring mechanisms being abused. We recommend a random string of letters at least 10 characters long. e.g. "urcajfielsjfs"

Syntax: g_mirror_secret string

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_threads - Max threads we can use during resync_fast, default 6

During resync fast four threads are used, this is usually sufficient, more may overload your system and result in failures, if your system is not under load you could set it as high as eight, but this would only be sensible if your disk array has more than 4 drives in it!

Syntax: g_mirror_threads int

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_trash](#), [g_mirror_debug](#)

g_mirror_trash - Normally on a resync the trash folder is ignored.

This can be useful when you want to compare results so you want everything even if it's a bit pointless

Syntax: g_mirror_trash bool

See also: [g_mirror_host](#), [g_mirror_nossl](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_mirror_mode](#), [g_mirror_secret](#), [g_mirror_prune_age](#), [g_mirror_threads](#), [g_mirror_live](#), [g_mirror_live_max](#), [g_mirror_config](#), [g_mirror_config_except](#), [g_mirror_debug](#)

g_monitor_disable - Disable the monitor process

This allows the monitor process to be completely disabled. The monitor process is the swatch executable and can be setup to monitor and automatically restart SurgeMail if it crashes. The monitor process is also used to start SurgeMail from the using the web interface if it has been shutdown.

Syntax: g_monitor_disable bool

g_monitor_port - SurgeMail monitor port (default 7027)

The port SurgeMail monitor runs on allowing SurgeMail to be remotely started. Typically you won't need to change this, however you can specify an IP address to bind to or a list of alternate ports, e.g. 10.3.2.3:7027 or 7027,8027 etc...

Syntax: g_monitor_port int

g_msg_hops_max - Maximum received lines or message is bounced, default 30

If there are more received lines than this the message is bounced.

Syntax: g_msg_hops_max int

g_msg_log_extra - Extra user activity logging

Log user activities like logins (successful and failed), and the download of message bodies to the 'msg.log' files; recYYYYMM/msgYYYYMMDD.rec

Syntax: g_msg_log_extra bool

g_msg_max - Max size of a single message

Max size, in bytes, of a message, eg: 20,000,000 for a 20mb limit. This setting is useful to prevent a single large message jamming up your system.

Syntax: g_msg_max int

g_msg_max_drop - Drop link if size exceeded instead of waiting for the message to all arrive

This setting has no further documentation currently available

Syntax: g_msg_max_drop int

g_msg_max_total - Max size of a message * recipients

This limits abuse, if set to 100mb then if user sends 10mb message to 10 users it will be blocked

Syntax: g_msg_max_total int

g_msg_track - Message tracking - for debugging

Debugging setting, do not use

Syntax: g_msg_track bool

g_mutex_fast - Use fast mutex handling DEBUGGING option only

Internal use only

Syntax: g_mutex_fast bool

g_mutex_timing - Name of mutex to collect extra timing information for

Internal use only

Syntax: g_mutex_timing string

g_mx_tryall - Try all mx hosts even if lower than own mx priority

This breaks the standard RFC behavior, but can be sensible in certain rare situations which currently escape me.

Syntax: g_mx_tryall int

g_myrbld_store - Size of internal myrbld database

Best not to touch this setting, default is 10000, Suggested valid range would be no less than 1000 and no more than 100000

Syntax: g_myrbld_store int

g_myrbld_testing - Testing internal database(do not use)

Testing myrbld code... still in development

Syntax: g_myrbld_testing bool

g_myrbld_to - Testing internal database(do not use)

Testing myrbld code... still in development

Syntax: g_myrbld_to string

g_naked_msg - Text to display if message body contains naked LF characters

Default is: "Naked LF see <http://netwinsite.com/surgemail/help/smtplf.htm>"

Syntax: `g_naked_msg` string

g_newui_advanced - Always run new admin ui in advanced mode

This setting has no further documentation currently available

Syntax: `g_newui_advanced` bool

g_newui_disable - Disable new admin ui (do not use)

This setting has no further documentation currently available

Syntax: `g_newui_disable` bool

g_notag_notascii - Don't add x-notascii: charset to any non ascii message

This can be used by user exception rules for users that don't expect any foreign language messages

Syntax: `g_notag_notascii` bool

g_notag_url_forgery - Don't add x-UrlForgery when a ref urls seem to not match

Many scam's will use legit urls with aref links to their own site, this tries to tag such messages which can then be scored as spam via `aspm_mfilter.rul`

Syntax: `g_notag_url_forgery` bool

g_old_imap_headbody - Get head and body seperately

This is just the way it used to do it, I can't see any good reason for it, but I'm leaving this setting incase there is a reason :-)

Syntax: `g_old_imap_headbody` bool

g_old_pophost_debug - Log extra info when doing old pophost logins

Log extra info when doing old pophost logins for debugging.

Syntax: `g_old_pophost_debug` bool

g_old_user_check - Disable the account status enabled check on rcpt lines

Normally the account status field is checked at the recipient stage, this setting disables this check.

Syntax: `g_old_user_check` bool

See also: [g_allow_user_authent_field_get](#), [g_allow_user_authent_field_set](#), [g_authent_always](#), [g_authent_any](#), [g_authent_allow_badascii](#), [g_authent_prefix_sep](#), [g_authent_process](#), [g_authent_cachelife](#), [g_authent_cachebad](#), [g_authent_cachesize](#), [g_authent_domain](#), [g_authent_encrypt_key](#), [g_authent_number](#), [g_authent_info](#), [g_authent_info_grp](#), [g_authent_ip](#), [g_authent_path_broken](#), [g_authent_single](#), [g_authent_spaces](#), [g_authent_strip_domain](#), [g_authent_restart](#), [g_authent_logall](#), [g_authent_fwdfile](#), [g_authent_timeout](#), [g_authent_last_login](#)

g_orbs_cache_life - Sets the amount of time to keep RBL entries cached.

Syntax: `g_orbs_cache_life` "seconds"

Default: 7200 seconds

This allows you to control how long the RBL lookups are cached for.

Example:

`g_orbs_cache_life "100"`

Syntax: `g_orbs_cache_life` int

g_orbs_check_all - Keep doing lookups even if found in a RBL, this is slower of course!

This checks all the RBL servers listed even if the connecting ip address is found in one server, this is slower but can mean you can score more accurately when an ip is listed in multiple RBL databases. Do not use with `g_orbs_late`, the two settings conflict and will not work. (`g_orbs_late` will be ignored)

Syntax: g_orbs_check_all bool

See also: [g_honeypot_rbl](#), [g_myrbt_testing](#), [g_myrbt_to](#), [g_myrbt_store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl_reject](#), [g_surbl_whois](#), [g_surbl_skip](#), [g_surbl_skip_ip](#)

g_orbs_exception - Exceptions to Open Relay / Known Spam sites

This allows you to over-ride a response from an ORBS/RBL database. For example, if a site you wish to do business with is in the RBL database you can add their IP address to this setting and then they can send you Email again.

Syntax: g_orbs_exception string

g_orbs_fake - Ip address to pretend we find in rbl database for testing

This setting has no further documentation currently available

Syntax: g_orbs_fake string

g_orbs_force - Forces RBL lookup even if they are in an exception.

Syntax: g_orbs_force "true/false"

This allows you to force RBL lookups on users that would normally not be checked due to being in an allowed relay ip (g_allow_relay_ip).

Syntax: g_orbs_force bool

g_orbs_late - Disconnect user only if they fail to authenticate

Sometimes your customers will be using dial in lines that are banned by RBL databases, in this situation this setting will help as it will keep the connection alive long enough for a valid user to send an smtp authentication in.

Can also be used with [g_spf_skip_to](#) "user@domain" this will allow you to add exceptions for users or domains that do not want RBL checks done on their accounts.

Syntax: g_orbs_late bool

g_orbs_list - Multiple Open Relay Blocking System RBL databases

Allows enforcement of a servers blacklisting or whitelisting in one or more RBL databases with a different action for each database. In addition this can be used to mark messages with a header which can then be taken into account in the SmiteCRC "SpamDetect rating" calculation. A RBL database is simply a DNS server that returns a positive response if a server is listed in the database. A variety of services are available online that can maintain blacklist databases. Normally you would maintain your own whitelist database that overrides the blacklist listings.

name=service action=deny,accept,stamp stamp="string to add to header ||remoteip||"

Where the stamp option adds the header:

X-ORBS-Stamp: string to add to header 1.2.3.4

The variable ||remoteip|| can be used to create a url to go directly to a spam database web site and give details on the offending ip address. e.g. stamp="Spamcop, http://spamcop.net/w3m?action=checkblock&ip=||remoteip||"

eg 1 - A simple deny mail from blacklisted servers could be achieved with:

```
g_orbs_list name="relays.ordb.org" action="deny"
```

eg 2 - A smarter setup with exceptions for certain IP ranges and a whilelist exception database, a blacklisted deny database and with useful header based tagging could be achieved as follows:

```
g_orbs_exception "127.0.0.*,12.34.56.*"
g_orbs_list name="mywhitedatabase.none" action="accept"
g_orbs_list name="relays.ordb.org" action="deny"
g_orbs_list name="relays.osirusoft.com" action="deny"
g_orbs_list name="bl.spamcop.net" action="stamp" stamp="spamcop, http://spamcop.net/w3m?action=checkblock&ip=||remoteip||"
```

eg 3 - To use the output of header based ORBS stamping in the SmiteCRC calculation the following could be used:

```
g_orbs_list name="relays.ordb.org" action="stamp" stamp="open relay"
g_orbs_list name="my.dialup.databse.none" action="stamp" stamp="dialup"
```

These entries have the following rules in filter.rul. If you used your own stamp text you would place appropriate entries in the local.rul file.

```
if(rexp_case("X-ORBS-Stamp", "open relay")) then
```

```
call spamdetect(4.0, "Sender's IP was on an open relay RBL")
endif

if(rexp_case("X-ORBS-Stamp", "dialup")) then
call spamdetect(4.0, "Sender's IP was on a dialup RBL")
endif
```

Some RBL lists return a numeric code to give extra meaning, for example 127.0.0.4 might mean an open relay, and 127.0.0.5 might mean the site has no postmaster address. You can specify multiple stamp messages using this format, stamp="4=Open Relay~5=No postmaster address~Default message goes here"

See Also: [RBL's](#)

Syntax: g_orbs_list name=string action=string stamp=string

g_orbs_rec - Log to record file if orbs deny action occurs

Log to record file if ORBS deny action occurs (can fill logs up).

Syntax: g_orbs_rec bool

g_orbs_report - List of IP's to check in RBL(s)

Use this setting to test your own ip addresses, as soon as one is found in a RBL you will be sent an email to alert you. The test is performed hourly. To test add 127.0.0.2 to the comma seperated list

Syntax: g_orbs_report string

g_orbs_service - Open Relay Blocking System RBL, service name (superceeded by g_orbs_list)

Set the name of the RBL service you want to use. A RBL service is a DNS database that has a record of all known spamming sites. If the server finds the connecting users IP address in this database all Email from their system is rejected. Also see the setting g_orbs_exception. Here are a few known RBL services, some charge and some are free!

- www.ordb.org
- inputs.orbs.org

Syntax: g_orbs_service string

g_orbs_submit - Do orbs check when 'data' command is sent or first valid recipient

This can reduce orbs lookups as a message that has no valid recipients will not trigger any orbs lookups.

Syntax: g_orbs_submit bool

g_orbs_system - Use system DNS lookups instead of SurgeMails for ORBS (not recommended)

If true use system DNS lookups instead of surgemails for orbs (not recommended).

Syntax: g_orbs_system bool

g_orbs_testing - ORBS testing

If true ORBSlookups are recorded but not blocked.

Syntax: g_orbs_testing bool

g_orbs_timeout - Orbs timeout

ORBS lookup timeout in seconds (default=10). If the timeout is reached the message is accepted and the failure is logged to mail.log.

Syntax: g_orbs_timeout int

g_perflog_disable - Disable perflog logging

Completely disable the logging of historica performance data for the status graphs.

Syntax: g_perflog_disable bool

g_perflog_flush_interval - Flush interval

Interval in seconds to flush the performance log files to disk. Default is 3600 s (ie once per hour)

Syntax: `g_perflog_flush_interval int`

g_perflog_logall - Log all counters

Log all counters including the currently undisplayed counters. This is useful if in the future you suddenly think, Oh I would really like to see the historic information on one of the undisplayed counters - which would normally not have been logged to file.

Syntax: `g_perflog_logall bool`

g_perflog_lowres - Log in low resolution

Normally data is logged every 10 seconds and 5 display scales are available hour, day, week, month and year. If this is set samples are taken every 5 minutes and 4 display scales are available: day, week, month, year.

Syntax: `g_perflog_lowres bool`

g_perflog_surgeononly - Only log surgemail counters

On Windows systems surgemail's performance logging will gather counters from surgemail and from the system "Perfmon" performance logging. This disables the collection of system counters.

Syntax: `g_perflog_surgeononly bool`

g_pipelining - Show pipelining in ehlo response

Show pipelining in ehlo response - not recommended - has no behavior affect.

Syntax: `g_pipelining bool`

g_pop_add_size - Improves pop performance on nfs slightly

This renames inbox messages to include the size of the file so that an lstat call is not needed.

Syntax: `g_pop_add_size bool`

g_pop_blocksize - Size of packets to read POP messages (best left alone)

Size of packets to read POP messages (best left alone).

Syntax: `g_pop_blocksize int`

g_pop_delay - Send POP packets after waiting for more data to send

This setting replaced `g_pop_nodelay`, as the default has been changed. It was changed as this can improve performance.

Syntax: `g_pop_delay bool`

g_pop_flush_lines - Flush to tcp every line of message sent (slow)

Too debug faulty network/client pop issues, not for general use, this may slow performance significantly

Syntax: `g_pop_flush_lines bool`

g_pop_lock - Lock out duplicate POP users with the file system

Use this setting if you are sharing a file system between multiple mail servers. This will make the mail server lock the users files to prevent a second user of the same name logging in and reading mail from one of the other systems.

Syntax: `g_pop_lock bool`

g_pop_max - Max total POP & IMAP users at any one time

This limits the channels that will be used at any one time for incoming POP and IMAP connections. The purpose of this setting is to prevent a sudden burst of users reading mail from using up all available channels. Generally setting this is a bad idea as there is a sensible default (dependent on the system resources available).

[See FAQ section on session limits](#)

Syntax: `g_pop_max string`

g_pop_min_late - Give min time error on first command after login

This may be less disruptive as it stops the client thinking the password is wrong.

Syntax: g_pop_min_late bool

g_pop_min_msg - Additional warning to give user when they login too soon

This lets you explain to the user what the problem is. Don't get carried away some clients may not like a long string here!

Syntax: g_pop_min_msg string

g_pop_min_skip - Skip ip addresses matching this list.

Useful for whitelisting webmail servers etc. 127.0.0.1 is always skipped

Syntax: g_pop_min_skip string

g_pop_min_time - Min time in seconds between consecutive POP logins, NEVER USE

If a pop client connects more often than this, give an error. This setting will very likely break webmail sessions and cause odd problems, Best avoided!

Syntax: g_pop_min_time int

g_pop_nolock - Allows concurrent pop logins, recommended

This setting avoids problems when users use pop and imap access to the same account at the same time.

Syntax: g_pop_nolock bool

g_pop_port - Port to listen for POP connections (default 110)

Typically you won't need to change this, however you can specify an IP address to bind to or a list of alternate ports, eg: 10.3.2.3:110 or 110,6110 etc... By default the mail server listens to port 110 on all adapters/addresses. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_pop_port string

g_pop_secure_port - Port to listen for secure POP connections (default 995)

Dedicated secure port to listen on for POP connections. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_pop_secure_port string

g_pop_warning - Send manager warning if this many sessions (pop or imap) reached (max 1 per hour)

This setting has no further documentation currently available

Syntax: g_pop_warning int

g_popfetch - Fetch incoming mail from another POP server

POPfetch will retrieve mail from POP accounts on another server and store it locally. The POP fetch interval can be set using g_popfetch_interval. The parameters for this setting are host(required), user(required), pass(required) or localuser(required).

eg:

g_popfetch host="netwin.co.nz" user="marijn" pass="secret" localuser="marijn@anydomain.com"

Alternatively POPfetch is able to attempt local delivery based on headers. Delivery is attempted to "X-Rcpt-To:" with fallback of "To:" and "Cc:" headers. To enable this the local user needs to be defined as "*",userxxx". Fetched mail will be delivered as specified in the headers or if no valid user is identified in the header to the default user "userxxx".

Syntax: g_popfetch host=string user=string pass=string localuser=string disable=bool

g_popfetch_interval - Interval between POPfetch attempts

The interval (in seconds) between successive attempts to fetch mail from remote mailserver POP accounts (as per g_popfetch rules). (default is 5 minutes = 300)

Syntax: g_popfetch_interval int

g_popfetch_kick - POPfetch will try and open the link for 10 seconds, then retry, this should bring up ISDN lines.

If true then POPfetch will try and open the link for 10 seconds, then retry, this should bring up ISDN lines.

Syntax: g_popfetch_kick bool

g_popfetch_nodup - Drop duplicate messages

Drop duplicate messages based on "Message-id:" header.

Syntax: g_popfetch_nodup bool

g_ppd_port - POPPassD port (default 106)

Port to listen for POPPassD connections. Typically you won't need to change this, however you can specify an IP address to bind to or a list of alternate ports, eg: 10.3.2.3:106 or 106,6106 etc... By default the mail server listens to port 106 on all adapters/addresses. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_ppd_port string

g_proxy - Proxy mode (or mailhost)

This enables the SurgeMail proxy mode, using 'tohost="xxx"' received from the authentication to determine real host for SMTP/POP connections. Any incoming SMTP, POP or IMAP connections will be passed on directly to the specified server. This allows you to split a domain over several separate systems. This method is outlined in general terms [here](#).

To setup a proxy server system with 4 machines (2 proxy, 2 backend) use the following steps, lets assume your hosts are PROXY1, PROXY2, SERVER1, SERVER2

1) Set on the proxy servers in surgemail.ini **g_proxy "true"**

On the back end server use **g_pop_nolock "true"** (to avoid timing issues)

On the back end server set **g_tohost_local "server1"** (or server2) so it knows it's own name.

2) Configure your authent database to return 'tohost=xxx' for each user on your system, e.g. in nwauth

```
nwauth
set testuser1@test.com test tohost="SERVER1"
set testuser2@test.com test tohost="SERVER2"
lookup testuser1@test.com
+OK testuser1@test.com config 0 tohost="SERVER1"
```

3) Configure your load balancing router to send users to PROXY1 & PROXY2, ...

4) When new users are added always define the 'tohost' setting to define which system they are added to as load increases you can add more backend or frontend servers as needed.

This is very similar to the 'mailhost' setting some systems use in LDAPAuth to translate mailhost to 'tohost' you would use: info_fields mailhost,tohost in ldapauth.ini

Syntax: g_proxy bool

g_proxy_default - Default proxy host

Default host to forward to if 'tohost' is not defined in user database for this user.

Syntax: g_proxy_default string

g_proxy_to_gateways - Proxy pop/imap connections to matching gateway settings

This setting has no further documentation currently available

Syntax: g_proxy_to_gateways bool

g_proxy_webmail - Redirect user.cgi logins to external host name

This lets you use a front end server to move web based logins onto the correct webmail host

Syntax: g_proxy_webmail host=string redirect=string

g_pstat_disable - Disable pstat per user accounting (for debugging)

Used for debugging only, do not play with this.

Syntax: `g_pstat_disable` bool

g_queue_limit - If on disk queue exceeds this block incoming mail

If you send email in faster than it can be sent, the queue grows forever until the server fails due to huge directories or insufficient disk space, this setting stops the incoming messages so you are alerted to the problem before it becomes critical. Note that this stops all incoming mail, including local deliveries. This is the number of items

Syntax: `g_queue_limit` int

Example: `g_queue_limit "100000"`

See also: [g_queue_warning](#)

g_queue_max - Size of internal queue file cache

Size of internal mail queue file cache, range 500-3000.

Syntax: `g_queue_max` int

g_queue_warning - If on disk queue exceeds this send manager a warning

If you send email in faster than it can be sent, or something is wrong (e.g. a broken dns server) then this helps warn you early

Syntax: `g_queue_warning` int

Example: `g_queue_warning "10000"`

See also: [g_queue_limit](#)

g_quota - Disk quota for users in specified `g_access_group`

If the user is in the specified group they get the specified disk quota. This is applied if no quota is specified in the `authent` module.

Syntax: `g_quota group=string quota=string`

See also: [quota_default](#), [quota_domain](#), [user_sms_quota](#), [user_list_quota](#), [webdav_quota](#), [g_quota_warning_disable](#), [g_quota_rcpt_disable](#), [g_quota_try_later](#), [g_quota_friends](#), [g_quota_skip](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_quota_disable - Disable quota system

Disables quota processing completely

Syntax: `g_quota_disable` bool

g_quota_friends - Count stored spam as part of quota

Count friends pending messages and spam store as part of the per user quota.

Syntax: `g_quota_friends` bool

See also: [quota_default](#), [quota_domain](#), [user_sms_quota](#), [user_list_quota](#), [webdav_quota](#), [g_quota_warning_disable](#), [g_quota_rcpt_disable](#), [g_quota_try_later](#), [g_quota_skip](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_quota_rcpt_disable - Disables quota check at rcpt stage

SurgeMail now does quota checking at rcpt stage (Quota checking used to be done after data arrived) This setting disables the quota checking at rcpt stage if the above causes problems (not intended for general use).

Syntax: `g_quota_rcpt_disable` bool

See also: [quota_default](#), [quota_domain](#), [user_sms_quota](#), [user_list_quota](#), [webdav_quota](#), [g_quota_warning_disable](#), [g_quota_try_later](#), [g_quota_friends](#), [g_quota_skip](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_quota_report - Send quota warnings to the manager

Useful for small systems where any quota limit failure is an issue for the manager to resolve

Syntax: g_quota_report bool

g_quota_skip - Skip quota checks for matching ip addresses

Skips the quota checking. Use this if you have a high priority robot (like your billing system) that must be able to deliver email to users (or students) even if the user is over quota.

Syntax: g_quota_skip string

See also: [quota default](#), [quota domain](#), [user sms quota](#), [user list quota](#), [webdav quota](#), [g_quota_warning_disable](#), [g_quota_rcpt_disable](#), [g_quota_try_later](#), [g_quota_friends](#), [g_quota](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_quota_try_later - Retry responses for over quota

Give 450 response if user is over quota so message will be resent.

Syntax: g_quota_try_later bool

See also: [quota default](#), [quota domain](#), [user sms quota](#), [user list quota](#), [webdav quota](#), [g_quota_warning_disable](#), [g_quota_rcpt_disable](#), [g_quota_friends](#), [g_quota_skip](#), [g_quota](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_quota_warning_disable - Disables the 80% quota warning message

Disables the 80% quota warning message.

Syntax: g_quota_warning_disable bool

See also: [quota default](#), [quota domain](#), [user sms quota](#), [user list quota](#), [webdav quota](#), [g_quota_rcpt_disable](#), [g_quota_try_later](#), [g_quota_friends](#), [g_quota_skip](#), [g_quota](#), [g_quota_disable](#), [g_quota_report](#), [g_share_quota](#), [g_user_sms_quota](#), [g_user_list_quota](#)

g_rcpt_bang - Allow bang characters in addresses

Allow exclamation marks in addresses. ie '!'

Syntax: g_rcpt_bang bool

See also: [rcpt msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_colon - Allow colon characters in addresses

Allow colon characters in addresses. ie ':'

Syntax: g_rcpt_colon bool

See also: [rcpt msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_max - Max recipients per message, default is 1000

Max recipients per message, default is 1000, can only be lower than 1000.

Syntax: g_rcpt_max int

See also: [rcpt msg](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_max_in - Limit for recipients of untrusted channels, default g_rcpt_max

This limit is only applied to untrusted sessions (incoming mail)

Syntax: g_rcpt_max_in int

See also: [rcpt msg](#), [g_rcpt_max](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_msg - Invalid recipient response

Response given for invalid recipient errors message is prefixed by email address..

Syntax: g_rcpt_msg string

See also: [rcpt_msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_nodup - Ignore duplicate recipients to the same user

When enabled this prevents a message being delivered more than once to a single person, it's a fairly good setting to use and will get rid of some spam for people using fallback addresses.

Syntax: `g_rcpt_nodup bool`

See also: [rcpt_msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_quote - Allow quote character(s) in addresses

By default quotes are blocked at the SMTP level, this is because some of the authentic modules don't handle quotes in addresses so it's best not to let them through. There is no known reason for ever turning this setting on.

Syntax: `g_rcpt_quote bool`

See also: [rcpt_msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rcpt_trace - Add X-Rcpt-Trace headers

This will list all recipients in the message to facilitate tracing

Syntax: `g_rcpt_trace bool`

See also: [rcpt_msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#)

g_rdns_timeout - Timeout for reverse DNS lookups default is 30 seconds

Best set between 10 and 60

Syntax: `g_rdns_timeout int`

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_received_name - Name shown in received headers

Name shown as received "by" in the received headers this defaults to server name but can be specified if required:

eg "myservername"

```
Received: from netwin.co.nz (unverified [10.0.0.5])
  by myservername (SurgeMail 1.5f) with ESMTP id 1140619
  for <marijn@netwin.co.nz>; Fri, 07 Nov 2003 10:25:59 +1300
```

Syntax: `g_received_name string`

g_received_names - List of valid received names for incoming email

This list is used when processing `vanish_bad_bounces`, `vanish_virus_bounces` and `vanish_any_bounce`. It defines the valid received names to expect quoted in a properly formed bounce message for a message from this server/system.

Syntax: `g_received_names string`

g_received_skip - Don't write a received header for local trusted users

This setting can be used to hide sensitive local ip addresses from outgoing mail headers. This will make tracking abuse more difficult, we do not recommend using this setting generally.

Syntax: `g_received_skip bool`

g_received_skip_all - Skip local received header for messages that have non local recipients

Note that in the case of a message that is to a local and remote recipient, it will skip the headers for both, even though the desire is to skip them for the remote recipient only. This not quite right, ideally one should skip this for outgoing only but since the header is added at delivery time we thought this was close enough.

Syntax: g_received_skip_all bool

g_received_skip_spf - Skip spf received header for messages that have non local recipients

Note that in the case of a message that is to a local and remote recipient, it will skip the headers for both, even though the desire is to skip them for the remote recipient only. This not quite right, ideally one should skip this for outgoing only but since the header is added at delivery time we thought this was close enough.

Syntax: g_received_skip_spf bool

g_recent_bypass - Bypass recent login failure checking

This allows you to disable recent login failure checking for certain IP addresses. Normally there up to a maximum of 9 login attempts are allowed per connection.

Syntax: g_recent_bypass string

g_record_days - Period delivery logs are stored

The number of days SurgeMail message delivery logs are stored.

Syntax: g_record_days int

g_record_hash - Hash delivery logs

Message delivery logs may be stored in hashed format within g_record_path as <surgemail dir>\recYYMM\msgYYMMDD.rec

Syntax: g_record_hash bool

g_record_path - Path for mail delivery logs

Sets the path for the SurgeMail delivery logs. Delivery logs contain entries for mail received and delivered in a single file per day. See [Searching the Log Files](#) for more information.

Syntax: g_record_path string

g_redirect - Redirect messages to 'was' to the 'new' address

Specifies global redirection rule. These rules are applied to local and remote addresses so should be used with 'care', for domain based redirection use the redirect rules within a domain. An example rule would be: fred@xx.com --> bob@yy.com or *@xx.com --> joe@xx.com

Wild cards can be used and replaced, e.g.

```
g_redirect was="*@gadget.net" to="%1@gadget.com"
g_redirect was="*/*.gadget.com" to="%1-%2@gadget.com"
```

Would make

```
bob@gadget.net --> bob@gadget.com
fred@cool.gadget.com --> fred-cool@gadget.com
```

These rules are processed 'before' the domain is identified, therefore you cannot use host_alias domain values in them. Use a domain redirect rule if this is required.

You can also redirect a message to a robot or script like this:

```
g_redirect was="auto@mydomain.com" to="|/usr/local/myrobot.sh"
```

Your script can read the environment variables:

```
MAILFROM
RCPTTO
MSGSIZE
```

And must read the message on 'stdin', the message will be terminated with "crlf.crlf"

Your script can then process the message and if it want's to respond must use smtp to send a response back etc...

Your script will run as the user 'mail' so if that user does not have access to the script file or work files then it will fail :-)

Syntax: `g_redirect was=string to=string`

g_redirect_cc - Carbon Copy redirect message

Same as 'redirect' but the message is still delivered to the original address as well. For `g_redirect_cc` there are two special names defined "\$localdomain\$" and "\$remotedomain\$", which can be used in the 'was' parameter (requires SurgeMail 2.3).

Syntax: `g_redirect_cc was=string to=string`

g_redirect_from - Redirect message if from matches

Redirect a message to another address if the from matches.

Syntax: `g_redirect_from from=string to=string`

g_redirect_from_cc - Carbon Copy redirect message if from matches

Redirect a copy of the message to another address if the from matches still delivering to the original address as well.

Syntax: `g_redirect_from_cc from=string to=string`

g_redirect_hide - Hide the redirection in the SMTP output

Hide the redirection in the SMTP output

Syntax: `g_redirect_hide bool`

g_redirect_iflocal - If local domain, then apply redirect

This is for doing fancy redirection where the rule is only applied if the domain of the destination is a local domain. For example to redirect all messages to postmaster at any local domain to one particular admin user.

Syntax: `g_redirect_iflocal was=string to=string`

Example: `g_redirect_iflocal was="postmaster@*" to="john@main.domain"`

See also: [redirect](#), [redirect_max](#), [redirect_cc](#), [redirect_hash](#), [g_bounce_redirect](#), [g_redirect](#), [g_redirect_cc](#), [g_redirect_from](#), [g_redirect_from_cc](#), [g_redirect_hide](#), [g_redirect_ignore_errors](#), [g_redirect_noautocreate_rules](#)

g_redirect_ignore_errors - Accept email even if redirected addresses fail

We consider this to be faulty behaviour as it will lead to emails vanishing with no bounce, use entirely at your own risk.

Syntax: `g_redirect_ignore_errors bool`

g_redirect_noautocreate_rules - Don't create redirection rules for domains automatically

This will stop SurgeMail creating redirection rules for new domains such as postmaster,abuse and support

Syntax: `g_redirect_noautocreate_rules bool`

g_relay_allow_from - Allow relaying for known from addresses

This setting allows users to send outgoing Email if their envelope 'from' address is a known local address. This is a very bad idea in general as spammers can do this too. So in general don't use this setting except as a lesser of two evils. It will be detected by some open relay checking systems and your site can then end up listed as an open relay. If this happens your Emails will be rejected by other peoples systems. e.g.

`g_relay_allow_from "*"@my.domain,*@second.domain,fred@third.domain"`

Syntax: `g_relay_allow_from string`

g_relay_allow_ip - Allow relaying from these users

List the IP ranges of local users that you will allow to send 'OUTGOING' Email without using SMTP authentication, e.g. "127.0.0.1,10.0.*". In the past, mail servers used to permit this from any IP address, but since this was abused by 'spammers' all modern mail servers only allow this from known local IP addresses. Remote users should use 'smtp authentication' or login via POP protocol before sending Email, then SurgeMail will trust them. Do NOT set this to '*' If you do your system will be blocked as it will be assumed that spammers are using your system even if they are not!!!

Syntax: g_relay_allow_ip string

g_relay_dom_and_ip - Relay based on domain and IP

Allow relaying if the domain in the from envelope and IP address both match.

Syntax: g_relay_dom_and_ip domain=string ip=string

g_relay_ifnot - Accept locally only if not from this ip

This lets you send all email to 'mx' destination, even if the account is local, unless it is coming from a known ip address range.

Syntax: g_relay_ifnot string

g_relay_message - Message to display to users who try to relay

Text string displayed to users who try and relay.

Default (blank) is: "Relaying blocked, read new mail, add <sender.ip> to forwarding or enable smtp authentication in your mail client"

Syntax: g_relay_message string

g_relay_process - Relay process, e.g. testip.exe \$WHOIP, return 1 to allow relaying, 0=deny

Allows you to run an external program to lookup an ip address and decide if it is one of your users who should be allowed to relay. This can be used when your users login via some type of shared system so the ip ranges are not known but you do have a way of checking if a user of yours is 'currently' connected on an ip address

Syntax: g_relay_process string

Example: g_relay_process "c:/surgeemail/testip.exe \$WHOIP"

See also: [fallback relay](#), [lookup relay on from](#), [g_auth norelay](#), [g fallback relay if exists](#), [g from relay](#), [g from relay white](#), [g_relay allow ip](#), [g_relay allow from](#), [g_relay dom and ip](#), [g_relay window](#), [g_relay window from](#), [g_relay to](#), [g_relay to user](#), [g_relay ifnot](#), [g_relay message](#), [g smite skip relay](#), [g_spf rewrite relay](#)

g_relay_to - Relay to this domain from anyone

This setting allows mail from anyone to be relayed to the specified domain. The relaying is unconditional.

Syntax: g_relay_to string

g_relay_to_user - Relay to specific user from anyone

This setting has no further documentation currently available

Syntax: g_relay_to_user string

g_relay_window - Allow relaying after valid POP login

This sets the time after a valid POP login that you will allow a user on the same IP to send outgoing mail. In general it is safe to set this setting large and it can allow people using old mail clients (that do not know how to do SMTP authentication) to still send through your server without making your server an open relay.

Syntax: g_relay_window int

g_relay_window_from - Requires pop authed user is in from header of sent message
This must be used with g_relay_window, the matching is 'simplistic' and matches on the 'from envelope' but will stop most simple forms of abuse.

Syntax: g_relay_window_from bool

g_rename_files - Files to apply virus renaming to

Only takes effect if g_virus_rename is checked. Default is: "*.exe,*.pif,*.bat,*.com,*.cmd,*.jav,*.vbs,*.scr,*.wsh"

Syntax: g_rename_files string

g_report_host - Report facts to a central host
Not for general use currently

Syntax: g_report_host string

g_report_notspam - Testing internal database(do not use)
Testing myrbl code... still in developement

Syntax: g_report_notspam bool

g_report_spam - Testing internal database(do not use)
Testing myrbl code... still in developement

Syntax: g_report_spam bool

g_responder_delay - Delay between responses to the same address.

This setting has no further documentation currently available

Syntax: g_responder_delay string

g_responder_from - Send 'from' destination user
This improves delivery, but risks loops and other issues.

Syntax: g_responder_from bool

g_responder_safer - Only respond if the sender can be verified in some way (spf/domainkeys)
This setting makes the server less likely to be black listed by accidentally responding to a forged email.

Syntax: g_responder_safer bool

g_responder_utf8 - Send response in utf8 format
Allow utf8 chars in response

Syntax: g_responder_utf8 bool

g_restart - Auto restart server

If turned on Swatch (a spawned second process) checks every 30 seconds to see if the server is still running. If it isn't running but it's pid file still exists (so if it died) this second process restarts the missing server and sends the manager account an Email reporting the fault.

For this to work on NT you need to set Dr Watson NOT to show visual notification of faults:

```
This sets Dr Watson to be the default debugger)
c:/> drwtsn32 /i
This brings up the Dr Watson settings, un-tick "Visual Notification"
c:/> drwtsn32
```

Generally this setting is not needed and could be left off, but if an odd problem should develop, this setting can give you peace of mind for a few days while you wait for a problem resolution from NetWin.

Syntax: g_restart bool

g_retry_bounces - Max hours to keep trying to bounce messages

Max hours to keep trying to deliver a bounce the default is 48hrs

Syntax: g_retry_bounces int

g_retry_dns - Hours to keep trying if dns response suggested invalid domain name, default 0

By default, if the DNS server says a domain doesn't exist, the message is immediately bounced so the sending user can take action. In some rare cases this will occur with a valid domain name because the actual DNS of the domain you are sending to is temporarily down. In this situation making SurgeMail retry for 1 hour can prevent these false bounces. I don't recommend this setting as mostly the DNS response and cache etc is very very reliable because SurgeMail keeps a local cache of DNS lookups that worked on disk. So for a failure like this to occur it must be the first time the server has EVER looked up the domain, so the odds are extremely remote. Delaying a useful response to the user for 1 hour just for this remote chance is not wise in my opinion.

Syntax: g_retry_dns int

Example: g_retry_dns "1"

See also: [g_dns_paranoid](#), [g_dns_match_msg](#), [g_dns_noptr](#), [g_dns_noptr_skip](#), [g_dns_noptr_msg](#), [g_dns_nocache](#), [g_dns_cache_size](#), [g_dns_system](#), [g_dns_host](#), [g_dns_nlookup](#), [g_dns_require](#), [g_dns_translate](#), [g_dns_old](#), [g_dns_new](#), [g_spf_dns_timeout](#)

g_retry_from - Time to keep messages from these domains

This setting has no further documentation currently available

Syntax: g_retry_from domain=string hours=string

g_retry_limit - Max hours to keep trying to deliver messages

Every hour the mail server will attempt to deliver any messages that fail for a reason that may be a temporary fault (for example the destination mail server doesn't respond). This setting limits how long these retries continue for. The default is 48 hours (2 days).

Syntax: g_retry_limit int

g_retry_minutes - Time between attempted retries

Time in minutes that SurgeMail will try and resend a message that has failed to be delivered. (default = 60 minutes).

Syntax: g_retry_minutes int

g_retry_rule - Retry rules overriding g_retry_limit

Rules that allow you to specify the retry_limit in hours on a per destination domain basis.

Example:

g_retry_rule domain="test.com" hours="48"

That will make it keep retrying to send to the domain test.com for 48 hours.

Syntax: g_retry_rule domain=string hours=string

g_retry_unwarn - Send user sent on confirmation if warning sent

This complements the warning setting, so the user can see the message did eventually go through and after how long...

Syntax: `g_retry_unwarn` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_retry_warn` - Send user a warning if first send fails

I like this setting myself but it can confuse users as the first send attempt will often fail and the user will mis read the bounce and think it's failed completely. It does mean when a message is urgent the user gets told right away, instead of 2 days later, that there is a problem sending the message so for a business it's a nice setting to enable.

Syntax: `g_retry_warn` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_retry_warn_n` - Send user a warning if nth send fails

Similar to the above setting but this one reduces the false warnings as messages often fail on the first attempt

Syntax: `g_retry_warn_n` int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_route` - Wildcard route mail to specified server

Route messages matching particular wildcard "from address" and wildcard "to address" to specified server. This is not a gateway rule and is only applied to mail that has already been accepted via SMTP authentication, relaying rules or gateway rules.

This would typically be used to route all mail for a particular user on a domain to another mailserver or to route all mail from a local domain through another server:

Case 1: Route mail for one user to another server

```
g_route from="*@*" to="user@localdomain.com" dest="1.2.3.4" user="" pass=""
```

Case 2: Route all mail from local domain through other server

```
g_route from="*@localdomain.com" to="*" dest="1.2.3.4" user="" pass=""
```

`g_route_except` gets applied allowing you to prevent mail coming in from certain IP addresses to be routed.

Syntax: g_route from=string to=string dest=string user=string pass=string

g_route_by_tohost - Route based on authent 'tohost' field

Use routing to a particular server based on 'tohost' setting in authentication database. This is particularly useful if you have users spread over several physical locations and want to be able to route mail for different users to particular servers.

Syntax: g_route_by_tohost bool

g_route_except - IP exception to g_route and g_route_by_tohost

IP exception to g_route and g_route_by_tohost.

Syntax: g_route_except string

g_route_local - Route messages for local domains if the rule applies

This setting has no further documentation currently available

Syntax: g_route_local bool

g_safe_imap - Force users to prove they are real if logging in from pop/imap

This feature is intended to prevent spammers/hackers from harvesting accounts on your system and then using them to send out spam

Syntax: g_safe_imap bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_safe_smtp - Force users to prove they are real if logging in from unknown sources via smtp

This feature is intended to prevent spammers/hackers from harvesting accounts on your system and then using them to send out spam

Syntax: g_safe_smtp bool

See also: [blogs_max_per_user](#), [loginfails](#), [url_blogs](#), [g_acctlog_sum_inactive](#), [g_acctlog_authonly](#), [g_acctlog_noaliases](#), [g_alias_login_disable](#), [g_bad_login_mins](#), [g_bad_login_allow](#), [g_bad_login_ip_allow](#), [g_bad_login_ip_ignore](#), [g_bank_log](#), [g_authent_logall](#), [g_authent_last_login](#), [g_autologin_pop](#), [g_autologin_file](#), [g_autologin_imap_disable](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_donly](#), [g_blogs_image_optional](#),

[g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#), [g_imap_log_protocol](#), [g_imap_log_flush](#), [g_imap_log_copy](#), [g_imap_log_header](#), [g_imap_log_body](#), [g_last_login](#), [g_last_login_days](#), [g_log_flush](#), [g_log_fwd](#), [g_log_level](#), [g_log_disable](#), [g_log_path](#), [g_log_pid](#), [g_log_thid](#), [g_log_reject_disable](#), [g_log_bounce_disable](#), [g_log_dropped_disable](#), [g_log_tcp_read](#), [g_log_tcp_write](#), [g_log_norcpt](#), [g_log_size](#), [g_log_dns](#), [g_log_slow](#), [g_log_start_norotate](#), [g_log_user](#), [g_log_date](#), [g_log_date_msg](#), [g_log_syslog](#), [g_log_syslog_debug](#), [g_log_syslog_only](#), [g_log_syslog_host](#), [g_msg_log_extra](#), [g_setpassword_firstlogin](#), [g_perflog_disable](#), [g_perflog_flush_interval](#), [g_perflog_lowres](#), [g_perflog_logall](#), [g_perflog_surgeonly](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_spawn_log](#), [g_spf_debug_log](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_ssl_require_login](#), [g_surgeblog](#), [g_user_blogs](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#), [g_surgeweb_logall](#), [g_surgeplus_log_level](#)

g_sample_get - Sample account to check if deliveries work

The idea is to create several accounts on various public mail servers. Then send a test message using a mailing list or [g_redirect](#) rule to these test accounts, then use the command `tellmail sample_get CODE DELETE` to check if the messages have arrived. The first paramter of `tellmail sample_get` is a code it expects to find in the message headers (or subject) and the second paramter should be the keyword 'delete' if you want it to delete the sample messages.

Syntax: `g_sample_get host=string user=string pass=string`

g_sample_show - Headers to show from sample messages

Typicall you will list headers that are added by spam filters

Syntax: `g_sample_show string`

g_scan_action - Converts return value from g_scan_cmd to action on email

Converts return value from `g_scan_cmd`, action=drop,accept,bounce.

Syntax: `g_scan_action code=int action=string reason=string`

g_scan_cmd - Run command on message, and return integer

Run command on message, and return integer, see `g_scan_action`.

Syntax: `g_scan_cmd string`

g_sched_utoken_timeout - Timeout for sched utokens in minutes

Timeout for sched utokens in minutes.

Syntax: `g_sched_utoken_timeout int`

g_send_backoff - Backoff slow hosts

Seconds to leave slow responding host alone (default 900).

Syntax: `g_send_backoff int`

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_body_end_retry - Try again if connection fails after entire body sent

This setting will tend to result in 'duplicate' messages being received, so should not be used, but strictly speaking it is valid to retry in this situation, the trouble is the receiving mail server 'may' have a real copy of the message so may deliver it even though the connection was dropped.

Syntax: `g_send_body_end_retry` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_body_noretry` - Don't try and resend if failure during body send

By default SurgeMail retries to send messages if the tcp connection is lost during the body send part of sending an email message. In rare situations this may cause problems, for example while sending a large file if the receiving software is faulty and is dieing rather than responding with 'don't try again' error code. This behaviour was reversed before version 2.0h (e.g. it never retried)

Syntax: `g_send_body_noretry` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_body_once` - Don't try 3 times if failure occurs sending body

This setting disables the new feature where the server tries harder to deliver a message even if it 'might' result in duplicates being delivered.

Syntax: `g_send_body_once` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_conspeed` - Outgoing connections per second per destination, default is 4

This helps prevent surgemail exceed tarpit throttles common in unix mail servers, adjust at your own risk. This won't generally limit outgoing email speed so you don't need to touch it. A value of '1' means surgemail can make one connection each second.

Syntax: `g_send_conspeed` int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_delay` - Wait this many seconds after sending each item.

This is a simple throttle to limit sending speed to any single domain, a value of 2 seconds is probably reasonable. In general you would also set `G_SEND_MAX_PERDOM` to 1.

Syntax: `g_send_delay` int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_first_retry` - Minutes for first retry, default is 16 minutes, do not adjust!

It's best not to change this generally, if you set it too low then grey listing may fail, if you set it higher then email is delayed.

Syntax: `g_send_first_retry` int

See also: [send_helo](#), [g_ban_helo](#), [g_gateway_helo](#), [g_helo_optional](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_verify_helo](#)

`g_send_helo` - Domain to use for all outgoing SMTP helo commands

Fully qualified domain to use for all outgoing SMTP helo commands.

Syntax: `g_send_helo` string

See also: [send_helo](#), [g_ban_helo](#), [g_gateway_helo](#), [g_helo_optional](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_verify_helo](#)

`g_send_helo_from` - Use the sending domain for the helo command

If the senders domain name (in return path envelope) is a valid local domain, then it is used in the 'helo' command.

Not generally recommended. The correct use of the helo is to identify the sending machine, not the domain, so although this makes the headers look pretty it doesn't make them more correct in my opinion.

Syntax: `g_send_helo_from` bool

See also: [send_helo](#), [g_ban_helo](#), [g_gateway_helo](#), [g_helo_optional](#), [g_send_helo](#), [g_send_helo_in](#), [g_verify_helo](#)

`g_send_helo_in` - Lookup dns name of incoming ip connection on local interface

So this is the local ip name it looks up not the remote ip address name.

Syntax: `g_send_helo_in` bool

See also: [send_helo](#), [g_ban_helo](#), [g_gateway_helo](#), [g_helo_optional](#), [g_send_helo](#), [g_send_helo_from](#), [g_verify_helo](#)

`g_send_lines` - Send single line packets

Send messages in single line packets, slow! (for debugging)

Syntax: `g_send_lines` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_lowpriority - Ip address of bulk sending servers

This limits the impact from mailing lists that would otherwise clogg the server and prevent normal individual emails going through quickly

Syntax: g_send_lowpriority string

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_max - Max concurrent sending sessions

Maximum concurrent outgoing SMTP connections . You should not have to change this. The default is 100.

Syntax: g_send_max int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_max_perchan - Msgs to send on one open channel

This may help delivery if a server is incorrectly identifying your server as a spam source. A value of 1-5 would be reasonable

Syntax: g_send_max_perchan int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_max_perdom - Max concurrent sending sessions to a single domain

Maximum concurrent outgoing SMTP connections to a single domain. The default is 2. This can be set higher and the default used to be 6 however there are a few servers out there that don't like more than 2 channels being opened to them.

Syntax: g_send_max_perdom int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_max_rcpt - How many rcpt's to send per message when sending

Default is unlimited, Setting this to a small value like 10 may help some mail servers.

Syntax: g_send_max_rcpt int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfethc](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_no_domain - Message to show when domain points to us but can't find user or domain
Most useful when using **g_authent_always**, as this error will be shown to local users when sending to local users that don't exist.

Syntax: **g_send_no_domain** string

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_onpopfethc](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_nolimit - Don't apply **g_max_perdom** limit when sending to this domain
Use this on incoming mx servers for the local domain so it can use lots of channels to send the data through.

Syntax: **g_send_nolimit** string

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_no_domain](#), [g_send_onpopfethc](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_nopoll - Use sleep loop instead of poll (debugging only)
This is to try and find an elusive fault on some systems sending large emails, not for general use

Syntax: **g_send_nopoll** bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfethc](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_noskipslow - Don't skip slow hosts

Normally surgmail remembers hosts that are slow to open, fail and doesn't retry for 60 minutes.

Syntax: `g_send_noskipslow` bool

`g_send_onpopfetch` - Only send outgoing while doing a POPfetch

Only send outgoing while doing a POPfetch (For dialup use).

Syntax: `g_send_onpopfetch` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_open_timeout` - SMTP link open timeout

Timeout, in seconds when opening an SMTP link.

Syntax: `g_send_open_timeout` int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_retry_552` - Retry on 552 responses (typically quota exceeded)

Some faulty hosts return a 552 error when a user is over quota, this means that by the RFC SurgeMail must not try again to deliver the message. However this is clearly not a permanent error and so it's often wise to retry in this situation, This setting makes SurgeMail attempt retries when faced with this odd response.

Syntax: `g_send_retry_552` bool

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

`g_send_rewrite` - Rewrite envelope recipient at send stage, does not change destination server

This rewrites the recipient envelope, you can use wild cards, e.g. `*@this.domain %1@another.domain`, to rewrite 'from' addresses use `g_from_rewrite`

Syntax: `g_send_rewrite` was=string to=string

`g_send_speed` - max outbound bandwidth

Bytes per second to limit each outgoing channel to. eg: 10k

Syntax: `g_send_speed` int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_timeout - Send timeout

Timeout, in seconds when sending mail, default is 540 (9 minutes)

Syntax: g_send_timeout int

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_tolimit](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_send_tolimit - Limit speed to send to one or more domains.

Some large providers will assume you are a spammer if you send too many messages in an hour. If you have a large mailing list it's easy to break these limits, in which case some rules like this can prevent this problem.

Syntax: g_send_tolimit domain=string perhour=int

Example: g_send_tolimit domain="hotmail.com,*hotmail.com" perhour="60"

See also: [user_status_send](#), [user_send_max](#), [send_helo](#), [g_footer_send](#), [g_footer_sendonly](#), [g_send_first_retry](#), [g_send_helo](#), [g_send_helo_from](#), [g_send_helo_in](#), [g_send_backoff](#), [g_send_lines](#), [g_send_nopoll](#), [g_send_lowpriority](#), [g_send_max](#), [g_send_max_perchan](#), [g_send_max_perdom](#), [g_send_max_rcpt](#), [g_send_nolimit](#), [g_send_no_domain](#), [g_send_onpopfetch](#), [g_send_retry_552](#), [g_send_rewrite](#), [g_send_noskipslow](#), [g_send_speed](#), [g_send_conspeed](#), [g_send_delay](#), [g_send_timeout](#), [g_send_open_timeout](#), [g_send_body_noretry](#), [g_send_body_end_retry](#), [g_send_body_once](#), [g_user_status_send](#), [g_user_send_max](#), [g_user_send_rule](#), [g_user_send_warning](#), [g_user_send_ip](#), [g_user_send_white](#)

g_server_name - Wildcard "SERVER_NAME" translation for domain identification

The vdomain a user connects on is normally identified automatically for "user account self management" and for "webmail". In the event that the domain name is not the same as the host name (eg hostname = mail.domain.com, domainname = domain.com) the WebMail web server can automatically translate the SERVER_NAME variable.

This setting specifies a wild card list of URLs 'URL' with associated translated host name for "SERVER_NAME". If the URL matches then SERVER_NAME is set to the second part of this setting 'name'. eg: to host the domains domain.com and mail.domain.com on host mail.domain.com:

g_server_name url="*.domain.com" name="domain.com"

Note: If your server name is not the same as your domain name also check the per domain setting [URL host](#).

Syntax: g_server_name url=string name=string

g_server_stamp - Replaces SurgeMail and version string in "Received" headers

Replaces SurgeMail and version string in Received headers of process mail

Syntax: g_server_stamp string

g_setpassword_firstlogin - Accept any password on first POP login and set in database (EMERGENCY USE ONLY, requires nauth -reasonfail parameter)

This setting has no further documentation currently available

Syntax: g_setpassword_firstlogin bool

g_share_home - Allow sharing of home directory

This allows sharing of the home directory in the unlikely situation that you might want to run separate surgmail processes. eg one process to cope with SMTP and another to cope with POP access.

Syntax: g_share_home bool

g_share_mail - Allow sharing of mail directory

Set true if mail area is shared (by nfs or other mechanism)

Syntax: g_share_mail bool

g_share_quota - Do quota on disk (e.g. when using nfs shared spool)

Normally SurgeMail keeps track of quota for all users in memory, this is efficient, but means if your are using a shared mail spool the quota figures are completely wrong, so use this setting to make surgemail keep track of quota's on disk, it increases disk load a bit of course but not too much.

Syntax: g_share_quota bool

g_smite_all - Add smite headers to all messages passing through server

Normally SmiteSpam headers are only added for locally delivered messages. This setting to all messages passing through this server.

Syntax: g_smite_all bool

g_smite_gateway - Add smite headers to gatewayed messages

Normally SmiteSpam headers are only added for locally delivered messages. This setting adds the headers for gatewayed messages too. This also adds headers to messages that are redirected by forward rules as well.

Syntax: g_smite_gateway bool

g_smite_level - Smite level to discard message

If [SmiteSpam](#) gives a message a "smite score" above this, throw it awayl. This setting is best never used. If used it should be set to '1 or 2'. A value of 1 = "has been reported", 2 = "has been reported multiple times". If smite match score is above this drop message. This is applied when the user downloads the email not at delivery time. What you probably want is 'g_spam_bounce' described elsewhere on this page.

Syntax: g_smite_level int

g_smite_skip - Skip smitecrc processing for messages from these domains

This will skip running SmiteCRC for messages whose from address matches these domains. This is the mail from envelope header NOT the from header in the message (you can check the return path header in the message to check what you need to add for this setting).

Note this is a wildcard field so to match any mail claiming to be from safedomain.com you would have to set:

g_smite_skip "*"@safedomain.com"

Syntax: g_smite_skip string

g_smite_skip_auth - Skip spam scanner if user logged in

Skips spam checks and spam header generation for any authenticated local user.

Syntax: g_smite_skip_auth bool

g_smite_skip_ip - Skip smite based on sender IP

Skip smite scanner if sender IP matches this wild card list.

Syntax: g_smite_skip_ip string

g_smite_skip_relay - Skip spam scanner if ip can relay

Skips spam checks and spam header generation for any local user.

Syntax: g_smite_skip_relay bool**g_smite_skip_to - Skip smite based on <to>**

Skip smite scanner if to matches this wild card to <address>.

Syntax: g_smite_skip_to string

g_smite_tag - Tag message if in SmiteSpam database

If set to true will tag messages already in the [SmiteSpam](#) database. A value of 1 = "has been reported", 2 = "has been reported multiple times".

Syntax: g_smite_tag bool

g_sms_forward - Specifies IP's which are allowed to forward to SMS gateways

Normally sms gateways are restricted to authenticated users (SMTP authentication) this allows you to specify IP's which can send without authentication. For example you may want your dlist server to send SMS, in which case you might add 127.0.0.1 to this setting.

Syntax: g_sms_forward string

g_sms_gateway - Address and port of your SMS gateway

This is the ip and port of an 'email to sms gateway'. The gateway should accept SMTP messages on this port and convert the email into an sms message then deliver to the phone number in the 'to' address. SMSGate is our 'email to sms gateway' and is FREE with SurgeMail. Setting [user_sms](#) to "true" for a domain allows users to specify a phone number (or email address) and rules for when to notify them.

Syntax: g_sms_gateway string

g_sms_gateway_force - Force sms notifications to go to g_sms_gateway

If a user sets their sms number to an email address, perhaps to make use of an existing gateway, then surgemail will send the message to the domain in that address. If you set this you can force the email to go to g_sms_gateway. NOTE: It is possible to configure SMSGate with 'send_mode smtp', 'recv_mode none' and no GSM modem. In this setup it simply reformats messages passing them on to the configured smtp_outserver for delivery as email messages.

Syntax: g_sms_gateway_force bool

g_sms_gateway_msgbytes - Maximum amount of message to send to g_sms_gateway (bytes)

Defines the maximum number of bytes of 'body' text to send to the g_sms_gateway. All headers are sent, then the defined number of bytes of 'body' text. Defaults to 160. May be set larger than the default if you have a lot of html messages or multipart html and text messages. Should not be set too large as there is no point sending binary attachments and the like to an sms gateway.

Syntax: g_sms_gateway_msgbytes int

g_smtp_auth_debug - Auth Debug (do not use)

This setting has no further documentation currently available

Syntax: g_smtp_auth_debug bool

See also: [g_gateway_allow](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#)

g_smtp_auth_ip - Ip Addresses to accept smtp authentication from

This prevents a hacker sending out spam by cracking a users account details, users must login from an address specified in g_smtp_auth_ip or g_relay_allow_ip

Syntax: g_smtp_auth_ip string

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp auth off](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surge web backend smtp](#), [surge plus smtp server name](#), [g disable smtp after](#), [g dbabble smtp port](#), [g dbabble smtp prefix](#), [g deny smtp](#), [g safe smtp](#), [g manager smtp](#), [g smtp auth debug](#), [g smtp bounce nslow](#), [g smtp cmd timeout](#), [g smtp data timeout](#), [g smtp delay stamp](#), [g smtp welcome delay](#), [g smtp log protocol](#), [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nlimit](#), [g smtp max bad](#), [g smtp port](#), [g smtp port auth](#), [g smtp port force](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp help disable](#), [g smtp cram enable](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix no head](#), [g smtp thread](#), [g smtp auth off](#), [g smtp noauth](#), [g smtp noauth m](#), [g smtp noauth msg](#), [g verify smtp](#), [g surge plus smtp server name](#)

g_smtp_auth_off - Disable SMTP AUTH from unknown ip addresses

This prevents a hacker sending out spam by cracking a users account details, users must login from an address specified in g_smtp_auth_ip or g_relay_allow_ip

Syntax: g_smtp_auth_off bool

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp auth off](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surge web backend smtp](#), [surge plus smtp server name](#), [g disable smtp after](#), [g dbabble smtp port](#), [g dbabble smtp prefix](#), [g deny smtp](#), [g safe smtp](#), [g manager smtp](#), [g smtp auth debug](#), [g smtp bounce nslow](#), [g smtp cmd timeout](#), [g smtp data timeout](#), [g smtp delay stamp](#), [g smtp welcome delay](#), [g smtp log protocol](#), [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nlimit](#), [g smtp max bad](#), [g smtp port](#), [g smtp port auth](#), [g smtp port force](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp help disable](#), [g smtp cram enable](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix no head](#), [g smtp thread](#), [g smtp auth ip](#), [g smtp noauth](#), [g smtp noauth m](#), [g smtp noauth msg](#), [g verify smtp](#), [g surge plus smtp server name](#)

g_smtp_big - Slow down incoming SMTP reads to get bigger packets (experimental)

This setting tries to prevent thrashing by making the server slow down the speed it reads data in an attempt to get larger packets. This seemed to have no affect when I tested it, but play with it if you want, It is only intended to be useful when you have hundreds of incoming connections all very slowly sending in data, and the server is short of CPU.

Syntax: g_smtp_big bool

See also: [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nlimit](#), [g smtp max bad](#), [g smtp port](#), [g smtp port auth](#), [g smtp port force](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp fast bounce](#), [g smtp fix no head](#)

g_smtp_bounce_nslow - Number of handles to use for doing slow rejections of smtp connections

If external servers are over loading your server so much that it ends up in a cpu loop rejecting connections then increaseing this might help. But beware your system must not run out of file handles so don't set it too large, The default is 100

Syntax: g_smtp_bounce_nslow int

See also: [rcpt msg](#), [g rcpt max](#), [g rcpt max in](#), [g rcpt msg](#), [g rcpt bang](#), [g rcpt colon](#), [g rcpt quote](#), [g rcpt nodup](#), [g rcpt trace](#), [g smtp cmd timeout](#), [g smtp data timeout](#)

g_smtp_cmd_timeout - SMTP command timeout

Seconds to wait after getting a message for next command (workaround for sendmail bug)

Syntax: g_smtp_cmd_timeout int

See also: [rcpt msg](#), [g rcpt max](#), [g rcpt max in](#), [g rcpt msg](#), [g rcpt bang](#), [g rcpt colon](#), [g rcpt quote](#), [g rcpt nodup](#), [g rcpt trace](#), [g smtp bounce nslow](#), [g smtp data timeout](#)

g_smtp_cram_enable - Enable CRAM-MD5 authentication (requires nauth 4.0h or greater)

Please note that CRAM-MD5 does have security implications, specifically it means that the local users password must be stored in a semi reversable state in the authent database. Also you must be using the new version of the NWAauth module.

Syntax: g_smtp_cram_enable bool

See also: [smtp_auth_off](#), [surgeauth_auth](#), [g_acctlog_authonly](#), [g_allow_user_authent_field_get](#), [g_allow_user_authent_field_set](#), [g_authent_always](#), [g_authent_any](#), [g_authent_allow_badascii](#), [g_authent_prefix_sep](#), [g_authent_process](#), [g_authent_cachelife](#), [g_authent_cachebad](#), [g_authent_cachesize](#), [g_authent_domain](#), [g_authent_encrypt_key](#), [g_authent_number](#), [g_authent_info](#), [g_authent_info_grp](#), [g_authent_ip](#), [g_authent_path_broken](#), [g_authent_single](#), [g_authent_spaces](#), [g_authent_strip_domain](#), [g_authent_restart](#), [g_authent_logall](#), [g_authent_fwdfile](#), [g_authent_timeout](#), [g_authent_last_login](#), [g_auth_hide](#), [g_auth_norelay](#), [g_auth_skipgateway](#), [g_mirror_nauth](#), [g_mirror_nauth_always](#), [g_filter_pipe_nauth](#), [g_gateway_auth](#), [g_smtp_skip_auth](#), [g_smtp_auth_debug](#), [g_smtp_portauth](#), [g_smtp_etrn_auth](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauth](#), [g_smtp_noauthm](#), [g_smtp_noauth_msg](#), [g_spam_check_auth](#), [g_xauthuser_hide](#)

g_smtp_data_timeout - SMTP data timeout

Seconds to wait for SMTP data input.

Syntax: g_smtp_data_timeout int

See also: [rcpt_msg](#), [g_rcpt_max](#), [g_rcpt_max_in](#), [g_rcpt_msg](#), [g_rcpt_bang](#), [g_rcpt_colon](#), [g_rcpt_quote](#), [g_rcpt_nodup](#), [g_rcpt_trace](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#)

g_smtp_delay_stamp - Stamp message if sender doesn't wait for welcome

If true then if any smtp commands arrive before the 'helo' greeting is sent then a header is added to messages which will result in a higher spam score.

Syntax: g_smtp_delay_stamp bool

See also: [g_gateway_allow](#), [g_smtp_auth_debug](#), [g_smtp_welcome_delay](#)

g_smtp_etrn_auth - etrn if authenticatd

Only do etrn processing if user is authenticated.

Syntax: g_smtp_etrn_auth bool

See also: [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#)

g_smtp_fast_bounce - Reject bad connections immediately

Normally SurgeMail waits 1-10 seconds before rejecting a bad connection (rbl/limits,...), this reduces cpu usage and prevents some DOS attacks, this setting disables this behaviour.

Syntax: g_smtp_fast_bounce bool

See also: [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fix_nohead](#)

g_smtp_fix_nohead - Accept messages with no headers and try and cope

This setting tries to cope if the message contains no headers at all, it is not recommended of course but may be needed on occasion for bad scripts

Syntax: g_smtp_fix_nohead bool

See also: [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nolimit](#), [g smtp maxbad](#), [g smtp port](#), [g smtp portauth](#), [g smtp portforce](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#)

g_smtp_help_disable - disable smtp help command

Disable SMTP help command (minor security percaution).

Syntax: g_smtp_help_disable bool

g_smtp_log_protocol - Log SMTP protocol

If enabled, the SMTP protocol is logged to the mail.log file as "smtp: In" and "smtp: Out" entries.

Syntax: g_smtp_log_protocol bool

g_smtp_log_size - Size of smtp.log file

This sets the smtp.log file size, default is 2mb

Syntax: g_smtp_log_size int

See also: [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nolimit](#), [g smtp maxbad](#), [g smtp port](#), [g smtp portauth](#), [g smtp portforce](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix nohead](#)

g_smtp_max - Max total incoming SMTP connections

This limits the channels that will be used at any one time for incoming SMTP connections. The purpose of this setting is to prevent a sudden burst of spam from using up all available channels. Generally you do not need to change this. (Default = 250). Use the related setting **g_smtp_max_reason** to over-write the detailed error if you don't want spammers to know what your limits are set to.

Syntax: g_smtp_max int

See also: [g smtp log size](#), [g smtp warning](#), [g smtp max reason](#), [g smtp max nolimit](#), [g smtp maxbad](#), [g smtp port](#), [g smtp portauth](#), [g smtp portforce](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix nohead](#)

g_smtp_max_nolimit - IP based exceptions to g_smtp_max

This lets you specify IP based exceptions to g_smtp_max, so if you need a certain IP to open up many connections you would add that IP here.

eg. g_smtp_max_nolimit "10.0.0.50"

Syntax: g_smtp_max_nolimit string

See also: [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max reason](#), [g smtp maxbad](#), [g smtp port](#), [g smtp portauth](#), [g smtp portforce](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix nohead](#)

g_smtp_max_reason - Reason to give to user if g_smtp_max is exceeded

This is most useful when the host in question is being used for the wrong purpose (incoming when it's intended for outgoing etc), or simply to advise the user of a potential solution

Syntax: g_smtp_max_reason string

See also: [g smtp log size](#), [g smtp max](#), [g smtp warning](#), [g smtp max nolimit](#), [g smtp maxbad](#), [g smtp port](#), [g smtp portauth](#), [g smtp portforce](#), [g smtp secure port](#), [g smtp vrfy msg](#), [g smtp etrn auth](#), [g smtp no brackets](#), [g smtp big](#), [g smtp fast bounce](#), [g smtp fix nohead](#)

g_smtp_maxbad - Max bad SMTP commands

The maximum number of bad commands accepted per session before SurgeMail will drop the connection.

Example: `g_smtp_maxbad "10"`

Syntax: `g_smtp_maxbad int`

See also: [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#)

`g_smtp_no_brackets` - Allow from/rcpt without angle brackets

Some faulty mail clients forget to put the brackets <> around the recipient, this setting allows such faulty behavior. Not generally recommended.

Syntax: `g_smtp_no_brackets bool`

See also: [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#)

`g_smtp_noauth` - Accept incoming SMTP from these IPs (other IPs allowed if authenticated), default is *
Mail sent from other IP addresses is only accepted if user is authenticated. Typically used if your server is behind a firewall of some kind and should only allow incoming email from a particular IP address. Users will be able to send as from any IP address if they use smtp authentication.

Syntax: `g_smtp_noauth string`

See also: [disable_smtp_after](#), [old_smtpghost](#), [old_smtpghost_skip](#), [smtp_auth_off](#), [smtp_welcome](#), [smtp_welcome_name](#), [smtp_from_ip](#), [surgeplus_backend_smtp](#), [surgeplus_smtp_server_name](#), [g_disable_smtp_after](#), [g_dbabble_smtp_port](#), [g_dbabble_smtp_prefix](#), [g_deny_smtp](#), [g_safe_smtp](#), [g_manager_smtp](#), [g_smtp_auth_debug](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_help_disable](#), [g_smtp_cram_enable](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#), [g_smtp_thread](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauthm](#), [g_smtp_noauth_msg](#), [g_verify_smtp](#), [g_surgeplus_smtp_server_name](#)

`g_smtp_noauth_msg` - Message given when sender is told to use authentication because of `g_smtp_noauth`

Message sent to sender when they try and send to the server but are required to authenticate because of `g_smtp_noauth`

Syntax: `g_smtp_noauth_msg string`

See also: [disable_smtp_after](#), [old_smtpghost](#), [old_smtpghost_skip](#), [smtp_auth_off](#), [smtp_welcome](#), [smtp_welcome_name](#), [smtp_from_ip](#), [surgeplus_backend_smtp](#), [surgeplus_smtp_server_name](#), [g_disable_smtp_after](#), [g_dbabble_smtp_port](#), [g_dbabble_smtp_prefix](#), [g_deny_smtp](#), [g_safe_smtp](#), [g_manager_smtp](#), [g_smtp_auth_debug](#), [g_smtp_bounce_nslow](#), [g_smtp_cmd_timeout](#), [g_smtp_data_timeout](#), [g_smtp_delay_stamp](#), [g_smtp_welcome_delay](#), [g_smtp_log_protocol](#), [g_smtp_log_size](#), [g_smtp_max](#), [g_smtp_warning](#), [g_smtp_max_reason](#), [g_smtp_max_nolimit](#), [g_smtp_maxbad](#), [g_smtp_port](#), [g_smtp_portauth](#), [g_smtp_portforce](#), [g_smtp_secure_port](#), [g_smtp_vrfy_msg](#), [g_smtp_etrn_auth](#), [g_smtp_help_disable](#), [g_smtp_cram_enable](#), [g_smtp_no_brackets](#), [g_smtp_big](#), [g_smtp_fast_bounce](#), [g_smtp_fix_nohead](#), [g_smtp_thread](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_noauth](#), [g_smtp_noauthm](#), [g_verify_smtp](#), [g_surgeplus_smtp_server_name](#)

`g_smtp_noauthm` - Accept incoming SMTP from these IPs (multi line version of `g_smtp_noauth`), default is *

Mail sent from other IP addresses is only accepted if user is authenticated. Typically used if your server is

behind a firewall of some kind and should only allow incoming email from a particular IP address. Users will be able to send as from any IP address if they use smtp authentication.

Syntax: `g_smtp_noauthm` string

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp auth off](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surge web backend smtp](#), [surgeplus smtp server name](#), [g_disable smtp after](#), [g_dbabble smtp port](#), [g_dbabble smtp prefix](#), [g_deny smtp](#), [g_safe smtp](#), [g_manager smtp](#), [g_smtp auth debug](#), [g_smtp bounce nslow](#), [g_smtp cmd timeout](#), [g_smtp data timeout](#), [g_smtp delay stamp](#), [g_smtp welcome delay](#), [g_smtp log protocol](#), [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp help disable](#), [g_smtp cram enable](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#), [g_smtp thread](#), [g_smtp auth off](#), [g_smtp auth ip](#), [g_smtp noauth](#), [g_smtp noauth msg](#), [g_verify smtp](#), [g_surgeplus smtp server name](#)

g_smtp_port - Port to listen for SMTP connections (default 25)

Typically you won't need to change this however you can specify an IP address to bind to or a list of alternate ports, eg: 10.3.2.3:25 or 110,2110 etc... By default the mail server listens to port 25 on all adapters/addresses. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: `g_smtp_port` int

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_portauth - SMTP ports which require smtp authentication, typically 587

It is recommended (by some) that users send email to port 587, and it requires smtp authentication, and port 25 be blocked from client ip addresses to prevent viruses etc using email servers. Be sure to add ,587 to the `g_smtp_port` setting too!

Syntax: `g_smtp_portauth` string

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_portforce - Block logins for ports not listed in `g_smtp_portauth`

Use this to prevent local users logging into port 25, this also stops many spammers abusing your system as they will try and send on port 25

Syntax: `g_smtp_portforce` bool

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_secure_port - Port to listen for secure SMTP connections (default 465)

Port to listen on for dedicated SSL SMTP connections.

Syntax: `g_smtp_secure_port` int

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_thread - Use separate thread for incoming SMTP connections

This makes the server run a separate thread just to process incoming smtp connections, this can help on a

busy system to stop a huge load of smtp connections clogging up the pop/imap connection processing, it is rarely needed.

Syntax: g_smtp_thread bool

See also: [disable smtp after](#), [old smtp host](#), [old smtp host skip](#), [smtp auth off](#), [smtp welcome](#), [smtp welcome name](#), [smtp from ip](#), [surge web backend smtp](#), [surgeplus smtp server name](#), [g_disable smtp after](#), [g_dbabble smtp port](#), [g_dbabble smtp prefix](#), [g_deny smtp](#), [g_safe smtp](#), [g_manager smtp](#), [g_smtp auth debug](#), [g_smtp bounce nslow](#), [g_smtp cmd timeout](#), [g_smtp data timeout](#), [g_smtp delay stamp](#), [g_smtp welcome delay](#), [g_smtp log protocol](#), [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp help disable](#), [g_smtp cram enable](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#), [g_smtp auth off](#), [g_smtp auth ip](#), [g_smtp noauth](#), [g_smtp noauthm](#), [g_smtp noauth msg](#), [g_verify smtp](#), [g_surgeplus smtp server name](#)

g_smtp_vrfy_msg - VRFY response

Change Response to VRFY, e.g. 252 Not telling.

Syntax: g_smtp_vrfy_msg string

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp warning](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_warning - Send manager warning if this many sessions reached (max 1 per hour)

This setting has no further documentation currently available

Syntax: g_smtp_warning int

See also: [g_smtp log size](#), [g_smtp max](#), [g_smtp max reason](#), [g_smtp max nlimit](#), [g_smtp maxbad](#), [g_smtp port](#), [g_smtp portauth](#), [g_smtp portforce](#), [g_smtp secure port](#), [g_smtp vrfy msg](#), [g_smtp etrn auth](#), [g_smtp no brackets](#), [g_smtp big](#), [g_smtp fast bounce](#), [g_smtp fix nohead](#)

g_smtp_welcome_delay - delays welcome message

Syntax: g_smtp_welcome_delay "seconds"

This delays the welcome message sent by SurgeMail to a connecting server. If the server sends data to SurgeMail during this waiting time SurgeMail will drop their connection. The theory is that any well behaved server will wait for prompts and check them, but a lot of spamming software never takes any notice of prompts/responses and sends blindly. We believe a value of 1-3 seconds is ideal. You can also exempt ip's from this setting by using g_spam_allow "ip". Settings too high will cause real mail to be lost.

Examples:

```
g_smtp_welcome_delay "3"
g_spam_allow "127.0.0.1"
```

So above, delay giving the welcome message for 3 seconds, anyone that sends data in that 3 seconds will be dropped, but anything connecting from 127.0.0.1 will be able to send immediately (you should make sure webmail is exempt).

Syntax: g_smtp_welcome_delay int

See also: [g_gateway allow](#), [g_smtp auth debug](#), [g_smtp delay stamp](#)

g_spam_alias_any - User alias string e.g. "++" if defined then strip suffix from emails - not advised!

This allows each user an infinite number of aliases of the form user+extension@domain.name, this can cause problems so only enable with caution. Usually set to "++" but can be set to a single plus, but this will break any email address that contains a plus so not normally recommended. If used avoid defining it as a single character at least!

Syntax: g_spam_alias_any string

g_spam_allow - IP wild card of sites to exempt from spam limits

Typically use this to allow known mailing list servers that use your system to send messages in without being tarpitted. e.g. "127.0.0.1,local.ip.number". This same setting is an exception to the other spam rules.

Syntax: g_spam_allow string

g_spam_allow_disable - Disable allow bounce messages

Normally when SurgeMail detects an SPF failure it will give the sending an opportunity to send an email to a special address, If the sender does this then their IP address is permitted in future, this saves a lot of hassle generally, in rare situations you may not want this system, this setting will just simply bounce the message instead.

Syntax: g_spam_allow_disable bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spam_allow_known - Unblock IP address if we have received messages from it for 3 days (so it's not a transient spammer)

This setting makes the SPF strict settings much softer, basically it says any IP address we've known about for 3 days, is considered safe. This will still stop most spammers, particularly when used in combination with RBL lists which will block the 'repeat' offenders.

Syntax: g_spam_allow_known bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spam_allow_msg - Template for unblock messages, use ||reason|| and ||allow|| and maybe a url

This lets you tailor the 'allow' bounce message given to incoming messages that fail the SPF checks. ||reason|| becomes the reason for the failure and ||allow|| is either the allow email to send to, or a link to use (if using g_spf_byweb "TRUE").

Syntax: g_spam_allow_msg string

Example: g_spam_allow_msg "||reason||, to fix send an email to ||allow|| then resend original email."

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spam_allow_rbl - Give unblock message to RBL bounces too

This setting extends the 'allow' email system used by SPF to the RBL style of failures. This makes it much safer to use RBL lists in block mode instead of stamping mode. You really must have g_spam_block enabled for this setting to work, otherwise the 'allow' mechanism lets everything through so this becomes pointless :-)

Syntax: `g_spam_allow_rbl` bool

See also: [g_honeypot_rbl](#), [g_myrbt testing](#), [g_myrbt to](#), [g_myrbt store](#), [g_surbl](#), [g_surbl reject](#), [g_surbl whois](#), [g_surbl skip](#), [g_surbl skip ip](#)

g_spam_allow_recent - Exempt recent POP from spam limits

Skip spam rules if recent POP IP number (see `g_relay_window`).

Syntax: `g_spam_allow_recent` bool

g_spam_aspam - Aspam rating

Scale for Aspam default is 1.0. Valid range is zero to two.

The aspam matching based on it's database of known spam and non spam produces a score in the range -5 --> 5. The `g_spam_aspam` setting lets you 'scale' this score to increase/decrease the importance of the aspam rating. The result is then applied (added to) the spamdetect header.

Syntax: `g_spam_aspam` string

g_spam_autotrain - Autotrain "good" filter

Auto train spam filter good messages based on first 1,000 outgoing emails.

Syntax: `g_spam_autotrain` bool

g_spam_block - Block spam (as decided by spf etc), if not set then user or domain can set

This setting is critical, without it, all the spam is let through to the user, with it set to true, 95% of spam is blocked before it enters your server. So, generally you want this turned on, it should result in very few false positives as messages are 'grey list' bounced.

Syntax: `g_spam_block` bool

g_spam_block_gateway - Block spam gatewayed messages too

Use this setting on incoming mail servers or servers that relay to servers that implement SPF. Without this SPF blocking will not work as the back end server cannot perform the SPF checks/blocking.

Syntax: `g_spam_block_gateway` bool

See also: [g_friends allow spf](#), [g_friends spf fail bounce](#), [g_friends check spf](#), [g_received skip spf](#), [g_spf mode](#), [g_spf nocache](#), [g_spf rewrite](#), [g_spf rewrite relay](#), [g_spf norewrite](#), [g_spf dns timeout](#), [g_spf timeout](#), [g_spf domain](#), [g_spf user domain](#), [g_spf very strict](#), [g_spf debug log](#), [g_spf default](#), [g_spf default noblock](#), [g_spf skip](#), [g_spf skip from](#), [g_spf skip to](#), [g_spf rev skip](#), [g_spf share](#), [g_spf header](#), [g_spf baddns skip](#), [g_spf nogrey](#), [g_spf noallow](#), [g_spf required](#), [g_spf enforce local](#), [g_spflog enable](#), [g_spflog domains](#), [g_spf byweb](#), [g_spf web url](#)

g_spam_block_msg - Template for spf blocked message if allow is disabled

This error is given for SPF failures when the allow system is disabled. You are probably looking for the setting `g_spam_allow_msg`, as it is the one that is normally used when a user is 'blocked' by spf.

Syntax: `g_spam_block_msg` string

See also: [g_friends allow spf](#), [g_friends spf fail bounce](#), [g_friends check spf](#), [g_received skip spf](#), [g_spf mode](#), [g_spf nocache](#), [g_spf rewrite](#), [g_spf rewrite relay](#), [g_spf norewrite](#), [g_spf dns timeout](#), [g_spf timeout](#), [g_spf domain](#), [g_spf user domain](#), [g_spf very strict](#), [g_spf debug log](#), [g_spf default](#), [g_spf default noblock](#), [g_spf skip](#), [g_spf skip from](#), [g_spf skip to](#), [g_spf rev skip](#), [g_spf share](#), [g_spf header](#), [g_spf baddns skip](#), [g_spf nogrey](#), [g_spf noallow](#), [g_spf required](#), [g_spf enforce local](#), [g_spflog enable](#), [g_spflog domains](#), [g_spf byweb](#), [g_spf web url](#)

g_spam_body - Add SpamDetect header in body

If spamdetect score is above this, add spamdetect header at top of message body (in addition to the header). This allows mail clients that are not able to filter mail based on headers to filter out spam email. This can be set on a per user basis too. A value of 3 or 4 would be reasonable. The only real reason for this setting is some common mail clients are unable to scan non standard headers so cannot automatically file spam in a folder unless this is used. My recommendation is for such users to use the web interface to set actions individually.

Syntax: g_spam_body int

g_spam_body_more - Add more info to spam body (ip address, ptr address, reply to and bounce address)
This can help the user decide if the message really is spam

Syntax: g_spam_body_more bool

g_spam_body_url - Text part of info to add to body, usually a url to your site
On this page you should explain to your users why this tag was added to their message, and how they can adjust their spam settings etc.

Syntax: g_spam_body_url string

g_spam_bounce - Bounce local delivery based on spamdetect score

If spamdetect score (number of '*'s) is above this, bounce message if local delivery. e.g. 7 or 8 would be reasonable, 3 would be very strict, and less than 3 would certainly bounce real emails. I recommend you don't set this below 5.

Syntax: g_spam_bounce int

g_spam_bounce_all - Bounce local and remote delivery based on spamdetect score

If spamdetect score (number of '*'s) is above this, bounce message, this applies to all messages regardless of user settings. e.g. 7 or 8 would be reasonable, 3 would be very strict, and less than 3 would certainly bounce real emails. I recommend you don't set this below 5. This rule is applied as soon as the message is submitted, user spam settings do not override it.

Syntax: g_spam_bounce_all int

g_spam_bounce_text - Error text when message is bounced due to g_spam_bounce setting

As per description. Default is: "554 Failure Message looks like spam, sorry not wanted here q=311", where q is the message queue id.

Syntax: g_spam_bounce_text string

g_spam_bounce_trusted - If spamdetect score is above this, bounce message if trusted (spam_allow or authenticated)

Normally trusted users (spam_allow or smtp authenticated users) are never bounced due to spam content, this setting forces those users to also be checked for spam content.

Syntax: g_spam_bounce_trusted int

g_spam_catcher - Spam catcher addresses

Addresses on web pages that shouldn't get any email (robot bait), only for use with Asпам.
Any email going to the specified address will be sent to the issпам address for processing and the message will also be dropped. If the message has multiple rctp's and some are valid users, but one matches the catcher address, it is not delivered to anyone. If you need to enter a lot of spam catcher addresses then the best way is to just setup a single spam catcher address and then use [g_redirect](#) to redirect other addresses to the spam catcher address.

eg

g_spam_catcher "johnsmith@mydomain.com"

Syntax: g_spam_catcher string

g_spam_char - Character to use instead of '*' for smitesпам headers (best left alone if possible)

Changing this will cause no end of problems, so only do this when initially installing SurgeMail

Syntax: g_spam_char string

g_spam_check_auth - Enable spam rules for authenticated users

Normally authenticated users are exempt from spam rules when sending mail. This enables all spam checking rules for authenticated users.

Syntax: g_spam_check_auth bool

g_spam_cmd - Command line spam checker, use \$FILE\$ in cmd parameters

This allows you to run a simple external spam filter the return value is added as a header, X-SpamCmd: r=N, Is Spam/Not Spam, use local.rul file to translate this return value to a spam score. e.g. G_SPAM_CMD "snfrv2r3.exe xnk05x5vmipeaof7 \$FILE\$" if used with <http://www.armresearch.com/message-sniffer/>. If the program returns 0 then the words Not Spam are added, if the value is non zero then Is Spam is added, this makes filtering rules easier to add to local.rul, see <http://netwinsite.com/surgemail/help/spam.htm#external>

Syntax: g_spam_cmd string

g_spam_cmd_if - If internal spam rating is below this number, then run external filter

This allows you to only scan messages with an external filter if the message is not obviously spam

Syntax: g_spam_cmd_if int

g_spam_cmd_reject - If external filter returns number larger than this reject

Filters based on return code of external spam filter program

Syntax: g_spam_cmd_reject int

g_spam_cmd_skip - If internal spam rating is below this number, then skip external filter

This allows whitelisting to work

Syntax: g_spam_cmd_skip int

g_spam_content_disable - Disable asspam_content.txt rules

The file asspam_content.txt is fetched from netwinsite and used to identify certain common spam messages based on content. Each line in the file gives a list of words or phrases, if most of the words are found, then the rule matches. You can add your own rules to asspam_content_local.txt. In a message that matches a rule you will see in the spamdetect header, Content: cid=NNN cid=NNN, you can then match the NNN with the unique id of each rule in asspam_content.txt

Syntax: g_spam_content_disable bool

See also: [g_aspam_headers](#), [g_aspam_need_ip](#), [g_spam_aspam](#)

g_spam_flag - Add X-SPAM-FLAG: Yes header if smite score is above this level

Some filters and servers like to see this header, a good value for this might be 7. Valid range would be 1-15, with 1 marking almost everything as spam, and 15 marking almost nothing.

Syntax: g_spam_flag int

g_spam_folders - Train on any message dropped into the relevant folders

This allows a user to create two folders '-Train Is Spam-' and '-Train Not Spam-' and then run the asspam training mechanism by dropping messages into those folders

Syntax: g_spam_folders bool

g_spam_folders_show - List the special folders for all users

Without this setting the user must create the folder name correctly for training to work from imap folders

Syntax: `g_spam_folders_show` bool

`g_spam_from_blacklist` - Fetch list of bad domains to reject email from - not recommended

This feature fetches the file <http://www.sa-blacklist.stearns.org/sa-blacklist/sa-blacklist.current> and then uses it efficiently to block senders, it is a huge file (26mb). Not currently recommended, we don't think the hit rate of this filter method is high enough to be useful. url used is <http://www.sa-blacklist.stearns.org/sa-blacklist/sa-blacklist.current>

Syntax: `g_spam_from_blacklist` string

`g_spam_from_max` - Max outgoing messages per ipaddress/return path pair, 30 minutes, e.g. 5000

This limit is useful where a local machine is sending on behalf of many users without authentication and you want to limit potential abuse

Syntax: `g_spam_from_max` int

`g_spam_grey` - OBSOLETE DO NOT USE, Enable old greylisting for spf mechanism

The grey listing mechanism relies on the principle that spammers are not using real mail servers but using dumb robots that won't 'retry'. So if all incoming messages are asked to 'retry' then the spam will not be received but the non spam will get in eventually. This does create a delay on all incoming mail, and may stop some stupid mail servers from successfully delivering. I would tend not to use this setting myself.

Syntax: `g_spam_grey` bool

`g_spam_grey_bounce` - Bounce if message was allowed due to grey listing, and spam score is above this, default 8 (was 4)

Since messages which are allowed in due to grey listing generally can't accept friends bounces (as the sender is unverified) it's important to bounce them with an allow message instead if they look like spam

Syntax: `g_spam_grey_bounce` string

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

`g_spam_grey_classc` - Apply grey listing to x.x.x.*

In theory this broadens slightly what grey listing will accept.

Syntax: `g_spam_grey_classc` bool

See also: [g_spam_grey](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

`g_spam_grey_dflt` - Enable greylisting for spf default accept events (not recommended)

If a message is going to be accepted due to the spf default rule (so there was no real spf record), then this comes into play. If the message is not from a trusted person, or a domain that we have previously checked using grey listings. Then the message is bounced. If the sender then tries again to send the same message (from/to pair) within a few hours, but not within 1 minute, then that ip address is marked as 'good' and future messages from them are accepted. This setting will result in some real email bouncing but slightly reduce spam, we no longer recommend this setting.

Syntax: `g_spam_grey_dflt` bool

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#),

[g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_dflt_bad - Enable greylisting instead of allow in some cases (recommended for block or strict)

This setting enables grey listing for spf default failure events only, and only if it's the first message from that ip address if more arrive before the grey listing succeeds then allow bounces are sent instead

Syntax: g_spam_grey_dflt_bad bool

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_nofive - Skip 5-6 minute black window for these domains

Use this for domains that retry at 5 minute intervals, e.g. (*@cs.com,*@xyz.com), this skips a test used to detect a particularly virulent spammer who uses a robot that retries at exactly 5 minute intervals

Syntax: g_spam_grey_nofive string

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_nohard - Avoid hard spf bounces always try and do a grey list instead

This avoids the hard bounce you would normally get for failed real spf records.

Syntax: g_spam_grey_nohard bool

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spf_nogrey](#)

g_spam_grey_nseen - Number of messages from an unknown host, default is 6

When a host is unknown if it sends more than this many messages before the grey listing resend occurs then it's considered to be a spammer.

Syntax: g_spam_grey_nseen int

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_size - Size of grey listing table, default is 3000

On busy servers set this to a larger figure, e.g. 9000 so it can remember more grey listing events

Syntax: g_spam_grey_size int

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_verify - Skip grey listing if host was not listening

Skips the grey listing if the host didn't resend to the g_smtp_verify probe for g_spam_grey_dflt_bad

Syntax: g_spam_grey_verify bool

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_window](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_grey_window - Window to block bad messages, typically 60 seconds

This prevents a fast retry by a stupid robot, some robots now wait 5-6 minutes but some mail servers may retry that fast too :-)

Syntax: g_spam_grey_window int

See also: [g_spam_grey](#), [g_spam_grey_classc](#), [g_spam_grey_dflt](#), [g_spam_grey_dflt_bad](#), [g_spam_grey_verify](#), [g_spam_grey_size](#), [g_spam_grey_bounce](#), [g_spam_grey_nofive](#), [g_spam_grey_nseen](#), [g_spam_grey_nohard](#), [g_spf_nogrey](#)

g_spam_header_trust_ip - List of IP addresses from which to trust/accept existing X-SpamDetect headers in emails

Use this setting to specify the filter machines which perform spam scanning for this machine. Use this on the filter machine, to specify itself so that mailing list messages do not get scanning/tagged twice. Ensure your users are sending messages via the filter machine.

Syntax: g_spam_header_trust_ip string

g_spam_hold_hide - Hide spam hold settings for end users and other held2pend user.cgi tweaks

This setting has no further documentation currently available

Syntax: g_spam_hold_hide bool

g_spam_hold_keep - Spam hold timeout

How many days to store users spam hold messages before deleting them.

Default is 14 days.

eg. g_spam_hold "14"

Syntax: g_spam_hold_keep int

g_spam_info - Info line explaining aspm system

Info line and url to explain aspm system.

Syntax: g_spam_info string

g_spam_info_hide - Remove x-spamdetect-info header line

Removes the x-spamdetect-info header line.

Syntax: g_spam_info_hide bool

g_spam_internal - Enable internal Aspm spam processing system

Enable new 'internal' spam processing system, note this disables SmiteCRC too!

Syntax: g_spam_internal bool

g_spam_isspam_ignore - Don't block messages from ip addresses recorded as a spam source

This bounces all email from an address recorded as a spam source until it is recorded as a 'notspam' source, the blocking message allows the sender to bypass the block.

Syntax: g_spam_isspam_ignore bool

g_spam_isspam_kind - Allow isspam from recent pop, gateway to etc

Allow ASPAM training messages to (isspam) from any trusted source (e.g. any source that would be allowed to relay/send outgoing email). This setting is recommended.

Syntax: g_spam_isspam_kind bool

g_spam_lang - Add header with guess at body language

This adds a header which makes a best guess at the contents of the message, it should not be assumed to be 100 percent reliable! Also note that empty messages or messages containing only images may be classified as 'Unknown (English)'

Syntax: g_spam_lang bool

g_spam_notrain - Disable iss spam and notspam addresses

Disable iss spam and notspam addresses for user training.

Syntax: g_spam_notrain bool

g_spam_notspam - Spam collection address

Address that non authenticated users can send non spam to.

Example: g_spam_notspam "notspam@domain.com"

Syntax: g_spam_notspam string

g_spam_noupdate - Disable as spam updates

Disable fetch of as spam filter rules etc from netwinsite.

Syntax: g_spam_noupdate bool

**g_spam_phishing - Download list of known phishing addresses and block outgoing email to them
Use this to stop your users resonding via email to a known phishing address.**

Syntax: g_spam_phishing bool

g_spam_poly - Scale for poly word matching

Scale for poly word matching, default is 0.1, Valid range is zero to two, Use 1.0 to enable.

Syntax: g_spam_poly string

g_spam_poly_disable - Disable poly code.

Disables the poly statistical scoring feature which is part of As spam. Poly tries to analyze the frequency of word combinations in spam and not spam to identify if a message is likely to be spam or not. We don't consider the poly system to be very useful, it has two faults, it's behaviour is not 'understandable' and it is 'content based', SPF is a much superior system!

Syntax: g_spam_poly_disable bool

g_spam_private - Enable private email addresses for users to avoid spam

Note: The user will define these settings, after turning on this global setting the user can use the Web Self administration interface, press the 'Spam' button and the private email address is defined on that page.

This setting adds the ability for each user to create a private email address to bypass SPF/ Spam filters. The user would then typically increase the spam settings for their non private account to 'friends mode' and enable SPF. So only known friends will be able to contact them via the old address.

This allows the user to live 'spam free' without the risk of blocking email from real people.

The user must be careful with their new private address, it should only be used with humans, when entering an address in a web form or mailing list a special variant should be used e.g. user--from-WEBDOMAINNAME@users.domain

The user defines their private address, in the form user--PRIVATE@domain.com, e.g. if the users public address is joe@cool.com, and the user defines a private extension of "juggle" then the private address would be:

joe--juggle@cool.com

Email addressed to joe--juggle@cool.com is delivered without SPF or SPAM filtering / tagging.

In addition the user can enable 'from' matching which must look like this: username--KEYWORD-STRING@cool.com, the user specifies a keyword e.g. "match". Then anything addressed to the user in this form:

joe--match-STRING@cool.com

Will only be delivered if 'STRING' is found in the 'from' envelope address, otherwise it will bounce. So when entering an email address in a web page called "toys.com" the user would enter:

joe--match-toys@cool.com

Any -- extension that is not recognized will return a bounce suggesting they remove the extension and try again.

Syntax: g_spam_private bool

g_spam_probe - Probe suspect urls to find spammers

This setting searches email message from dodgy/unknown sources for urls, then looks at the page those urls refer to to see if those pages in turn point to a listed SURBL. Only domains matching a specific list of rules are scanned so there is almost no risk of this feature clicking on a page that might do something bad.

Syntax: g_spam_probe bool

g_spam_probe_more - Probe even if email is from a known ip address

Generally not advised

Syntax: g_spam_probe_more bool

g_spam_probe_unknown - Probe any unknown url (dangerous)

This setting increases the remote chance of probing a web page that might have some action (like a confirmation signup request, unsubscribe etc...), in practice there are a bunch of tests we perform so it would be most unusual for this problem to occur but it's safer not to use this option.

Syntax: g_spam_probe_unknown bool

g_spam_probe_whois - Do whois lookups on web pages found in probe

Some spammers register new domains each day, this probe checks the whois data to find if the new web site is owned by a known spammer

Syntax: g_spam_probe_whois bool

g_spam_share - Use and share some spam/aspam information with central server (netwin) experimental

This setting enables some features which let surgemail share information about spam and non spamming ip addresses with a central netwin server.

Syntax: g_spam_share bool

g_spam_status_hour - Process all spam status messages at this time (disk io intensive)

Normally the spam status emails are sent in response to incoming messages at undefined times, this allows all spam status emails to be sent at a predefined time.

Syntax: g_spam_status_hour int

g_spam_subject - Modify message subject line based on spam rating

If spamdetect score is above this add spam rating Spam:**** to subject.

Syntax: g_spam_subject int

g_spam_subject_dom - Destination domains to tag subject for

Note that g_spam_subject_gateway and G_SMITE_GATEWAY or G_SMITE_ALL must also be set to true for this to work. If this setting is blank then all gatewayed domains would get tagged. Tagging won't occur if

the message is not sent through a g_gateway rule or redirect rule

Syntax: g_spam_subject_dom string

g_spam_subject_gateway - Modify message subject line based on spam rating for gatewayed messages

If true then spam_subject setting applies to gatewayed messages too

Syntax: g_spam_subject_gateway bool

g_spam_subject_word - Allow arbitrary modification of message subject line

This is a string that is prefixed to the subject of incoming mail caught by g_spam_subject. You can use ||score|| and ||stars|| which will contain the actual spam rating. Good examples might be: "[SPAM]" or "SPAM(||score||), "

Syntax: g_spam_subject_word string

g_spam_url - Scale for url word matching

Scale for URL word matching, default is 0.3, Valid range is zero to two (recommend 1.0)

Syntax: g_spam_url string

g_spam_user_max - Max messages for authenticated users

Max messages an authenticated user can send per 30 minutes, eg: 5000

Syntax: g_spam_user_max int

**g_spam_user_skip - Users to skip g_spam_user_max limit for
Set this for special known users who send lots of email**

Syntax: g_spam_user_skip string

g_spam_userconfig - Enable per user spam settings

Allow users to opt in / out of specific anti spam features. If this is enabled this will add a "Spam" button on the users account self management pages.

The most useful antispam feature is that user's mail that is suspected spam, can be stored on the server so that these messages do not need to be downloaded to your normal email client over what could well be a low bandwidth connection.

Syntax: g_spam_userconfig bool

g_spam_vanish - Vanish local delivery based on spamdetect score

If spamdetect score (number of '*'s) is above this, vanish message if local delivery. eg: 12 would be reasonable.

Syntax: g_spam_vanish int

g_spam_vanish_all - Vanish local and remote delivery based on spamdetect score

If spamdetect score (number of '*'s) is above this, drop message, applies to all messages regardless of user settings. e.g. 14. This rule is applied as soon as the message is submitted, user spam settings do not override it.

Syntax: g_spam_vanish_all int

g_spamdetect_always - Always add spamdetect header

Always show spamdetect header even for low and negative scores.

Syntax: g_spamdetect_always bool

g_spawn_log - If true the spawns are logged to lib_spawn.log

Useful for finding obscure problems with spawned modules of various kinds, webmail, nwauth, virus

checkers etc.

Syntax: `g_spawn_log` bool

`g_spf_baddns_skip` - If spf dns failure then allow message through (instead of giving retry error)

This setting is not normally needed as lookups generate retry failures so the sending server tries again and the dns failure (which is usually temporary) won't occur the second time. Normally on a DNS failure SPF should give a 'retry' message, this is because spammers often have faulty DNS servers so that SPF checks always fail for their domain, so letting the message through will let some spam into your system. But in some situations the normal behavior might loose you real email so then using this setting at least until your dns problems are resolved might be wise.

Syntax: `g_spf_baddns_skip` bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

`g_spf_byweb` - Perform allow bounce confirmation via the web.

This gives a url to the sender in the allow bounce message instead of an allow-.. email address. The url gives a page with a verification image, the sender types the number seen and can then re-send their original email.

Syntax: `g_spf_byweb` bool

`g_spf_debug_log` - Enable spf.log file

By default this log is not generated as it's not usually needed.

Syntax: `g_spf_debug_log` bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

`g_spf_default` - (strict only) Default spf record if none found default 'mx/16 a ptr:%{d2} -all'

The example shown isn't entirely true, we adjust the 'd2' depending on the domain, so it's usually unwise to change this.

Syntax: `g_spf_default` string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

`g_spf_default_noblock` - (strict only) Only stamp headers if default spf record fails when no real spf header
This setting makes blocking occur only for REAL spf records, not for the default one applied to domains that have no SPF record defined.

Syntax: `g_spf_default_noblock` bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_dns_timeout - Seconds to wait for dns lookups for spf, best not to change
Generally a ten or twenty second timeout is reasonable. Adjusting the default is probably not necessary.

Syntax: g_spf_dns_timeout int

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_domain - Domain for SPF rewrite and allow messages (defaults to first domain on server)
When SurgeMail relays/forwards a message the 'from' address is rewritten (g_spf_rewrite should be true).
The new address is 'from' your domain and this setting tells surgemail which local domain to use for these from addresses.

Syntax: g_spf_domain string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_enforce_local - If spf fails and it's a local domain then skip grey listing and give allow message
This settings stops spammers who fake your own email domains, but it may upset users who are not authenticating or are using their own mail servers, so you will have to expect a few minor issues like that when you turn this on. This setting over-rides the 'users' spf and friends settings for local domains.

Syntax: g_spf_enforce_local bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_header - Use g_verify_mx_skip and apply to resulting ip
If the sending host matches g_verify_mx_skip, then spf tests are performed on the first received header not listed in that setting. Only stamping is possible though since this indicates a front end gateway and a reject would cause a 'bounce' which would not be safe

Syntax: g_spf_header bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),

[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_mode - Sender Permitted From

See <http://netwinsite.com/spf.htm> for details.

Syntax: g_spf_mode string

g_spf_noallow - Give hard bounce (no allow message) for spf failures for these domains & ignore friends
This toughens spf for critical domains (banks etc) where you don't want any forged messages leaking through. This setting over-rides the users spf/friends settings for these domains (so should be used with some caution)

Syntax: g_spf_noallow string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_nocache - Disable SPF cache

There is a small cache used for SPF results, This setting disables it.

Syntax: g_spf_nocache bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_nogrey - Skip SPF grey listing for these domains (require allow response)

This toughens spf for the domains in question, requiring that they really pass an 'allow' test rather than simply a grey listing test. Good for commonly forged domains which do normally obey spf.

Syntax: g_spf_nogrey string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_norewrite - Exceptions to rewrite rule, e.g. *@my.domain,bob@this.domain

Where you allow users to send through your server you may want to stop rewriting for their domains so that their from address is not munged. Local domains are automatically exempt from 'rewriting'. Specify *@domain.name not just domain.name

Syntax: g_spf_norewrite string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_dns_timeout](#), [g_spf_timeout](#),

[g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),
[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#),
[g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#),
[g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_required - Require an spf entry for these domains
Used to make select domains add spf to talk to you :-)

Syntax: g_spf_required string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#),
[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#),
[g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),
[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#),
[g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_enforce_local](#), [g_spflog_enable](#),
[g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_rev_skip - Skip SPF checks if reverse ip name matches in this list, e.g. *.yahoo.com
Where you identify a domain that does not support SPF and is often used in a manner which breaks SPF default rules this setting can safely allow the problem domain. This setting is probably not needed now most large mail systems are using SPF.

Syntax: g_spf_rev_skip string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#),
[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#),
[g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),
[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_share](#), [g_spf_header](#),
[g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#),
[g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_rewrite - Rewrite 'from' envelope in redirected mail (SRS)

When messages are redirected/forwarded to another server from you server, the 'from' address of the existing message envelope will no longer obey SPF rules as it will be coming from your server rather than the original server. So to fix this enable this rewrite setting and then the from envelope is rewritten to point to your system using a short life token. The 'from' header of the message is not modified.

Syntax: g_spf_rewrite bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#),
[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#),
[g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),
[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#),
[g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#),
[g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_rewrite_relay - Rewrite even if from ip is a host to relay for
In some cases you will want SRS rewriting for relay hosts, In which case you should turn this on.

Syntax: g_spf_rewrite_relay bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#),
[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#),
[g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#),
[g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#),
[g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#),
[g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_share - List of hosts to share allow ips with. Must all have same srs.secret file

List your other incoming mail servers (which must be running surgemail). This lets SurgeMail share the list of known IP addresses which have sent 'allow' emails. You must copy your srs.secret file across all of the servers in question so they can verify each other correctly.

Syntax: g_spf_share string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_skip - Skip spf checks for these ip addresses, e.g. other mx hosts

List the ip addresses of your other MX servers so SPF checks wont fail when a message comes in via an mx host instead of directly. The SPF checking must therefore be done on all the MX servers.

Syntax: g_spf_skip string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_skip_from - Skip based on from, e.g. noreply@*paypal.com,..., Also skips RBL

Good for skipping SPF checking if a domain is in some way incompatible with SPF checking

Syntax: g_spf_skip_from string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_skip_to - Skips SPF checks based on rcpt address and RBL checks.

Syntax: g_spf_skip_to "user@domain.com"

This setting can be used to skip spf checks based on the rcpt address, if used with [g_orbs_late](#) "true" then it can also be used to skip rbl checks if the rcpt matches this setting.

Syntax: g_spf_skip_to string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_timeout - Seconds to wait for all spf lookups to finish, default 48 seconds

Best not to change

Syntax: g_spf_timeout int

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#),

[g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_user_domain - Make allow bounces use destination user domain name

This can be useful if you need to ensure emails bounce with an address that is similar to the destination

Syntax: **g_spf_user_domain** bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_very_strict - (strict only) Only give 'allow' option for default spf rule failures not real ones

In this mode real SPF failures are hard failures, but if there is no SPF record for a domain then the friendly 'allow' system is used to let the user send mail with only mild difficulty.

Syntax: **g_spf_very_strict** bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spf_web_url - Specify full url for spf byweb commands http://domain.name:port

Normally the default will work.

Syntax: **g_spf_web_url** string

g_spflog_domains - Specify which domains should get spflog entries sent to them.

If some of your backend servers are not surgemail then this setting will be needed to turn off the spflog messages to the non surgemail servers

Syntax: **g_spflog_domains** string

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#), [g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_enable](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spflog_enable - Enable this if this server is a frontend for a SurgeMail server users log into.

Enable this if this server is a frontend for a SurgeMail server users log into.

Syntax: **g_spflog_enable** bool

See also: [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_received_skip_spf](#), [g_spf_mode](#), [g_spf_nocache](#), [g_spf_rewrite](#), [g_spf_rewrite_relay](#), [g_spf_norewrite](#), [g_spf_dns_timeout](#), [g_spf_timeout](#), [g_spf_domain](#), [g_spf_user_domain](#), [g_spf_very_strict](#), [g_spf_debug_log](#), [g_spf_default](#), [g_spf_default_noblock](#), [g_spf_skip](#), [g_spf_skip_from](#), [g_spf_skip_to](#), [g_spf_rev_skip](#), [g_spf_share](#),

[g_spf_header](#), [g_spf_baddns_skip](#), [g_spf_nogrey](#), [g_spf_noallow](#), [g_spf_required](#), [g_spf_enforce_local](#), [g_spflog_domains](#), [g_spf_byweb](#), [g_spf_web_url](#)

g_spool_path - Allows SurgeMail to scan a directory for messages to send.

Syntax: g_spool_path "directory of spool"

SurgeMail will scan this directory every few seconds and check for any messages in this directory if found SurgeMail will then send them the messages (must end in the extension .msg). The format of the messages is as follows (without the quotes).

filename: test.msg

"

To: you@domain.com

From: blah@domain.com

Subject: blah blah

This is a test

"

Syntax: g_spool_path string

g_ssl_allow - IP Wild card of connections to allow to use SSL

This setting controls which connecting IP numbers are permitted to use SSL on POP and IMAP. They will see TLS in the protocol extension command (ETRN for SMTP or CAPA for POP). Typically, to enable SSL you set this to "*" after getting a certificate. If you don't have a valid certificate then turning this on can cause problems as mail clients will try to use SSL and fail.

Syntax: g_ssl_allow string

g_ssl_allow_imap - IP Wild card list to allow SSL encryption from for imap

This setting controls which connecting IP numbers are permitted to use SSL on IMAP.

Syntax: g_ssl_allow_imap string

g_ssl_ciphers - List permitted ciphers

This can be used to enhance security, not recommended but is useful if you are trying to pass a security audit of some kind. A value of MEDIUM:HIGH is probably what you want to set it to. It is case sensitive.

Syntax: g_ssl_ciphers string

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossll](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_require_web](#), [g_ssl_try_out](#), [g_ssl_try_not](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_ssl_disable_sslv2 - Disable ssl 2.0 support for enhanced security, not recommended

Disables one of the older ssl protocols which slightly increases security and decreases compatibility with older clients

Syntax: g_ssl_disable_sslv2 bool

g_ssl_per_domain - Create/use an SSL certificate for each domain

SurgeMail can be set to use a single SSL certificate for the server or individual certificates on a per domain basis. Per domain SSL certificates can only be used with IP based vdomains.

SurgeMail will create private key / certificate pairs if required on startup. Alternatively these can be created using the 'SSL Config' link on the global settings page. These can be replaced with your own trusted signed certificates using the web admin interface or by placing the appropriate private key and certificate pem files in the following location: <surgemail>/ssl for a single certificate for the whole server and under <surgemail>/ssl/<vdomain> for per vdomain certificates.

Some mail clients and web browsers will complain if the certificate domain does not match the domain they are connecting to.

Changing `g_ssl_per_domain` will require surgemail to be restarted to take affect. Changes to certificates using the web admin interface now take affect immediately.

Syntax: `g_ssl_per_domain` bool

g_ssl_require - IP Wild card of connections to require to use SSL

This forces all matching IP addresses to use SSL for SMTP, POP and IMAP connections. Typically you would use this for non local connections to increase security local connections might be comparatively safe in un-encrypted mode.

Syntax: `g_ssl_require` string

g_ssl_require_imap - IP Wild card of connections to require to use SSL for IMAP

This forces all matching IP addresses to use SSL for IMAP connections.

Syntax: `g_ssl_require_imap` string

g_ssl_require_login - IP wildcard of connections for users needing to use SSL

This setting forces all matching IP addresses to use SSL for any action that requires a user login. eg: POP, IMAP and SMTP authentication but not plain SMTP. So this is ideal if you want all users to use SSL but still want email to come in from non SSL SMTP servers.

Syntax: `g_ssl_require_login` string

g_ssl_require_out - Other machines we only send to using SSL

This forces all matching IP addresses to use SSL for SMTP outgoing connections. Typically you would use this for outgoing connections to increase security.

Syntax: `g_ssl_require_out` string

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossl](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_web](#), [g_ssl_try_out](#), [g_ssl_try_not](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_ssl_require_web - Require https for most web features (excluding blogs file sharing and surgeplus)

This setting has no further documentation currently available

Syntax: `g_ssl_require_web` bool

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossl](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_try_out](#), [g_ssl_try_not](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_ssl_sha1_sign - Sign CSR with SHA1 instead of MD5 for enhanced security, beta testing

This will probably be made the default in the near future

Syntax: `g_ssl_sha1_sign` bool

g_ssl_try_from - Try and start ssl mode if from this user, e.g. *@xyz.com

Must also match the `g_ssl_try_out` rule, this lets you only do ssl when the email is 'from' certain domains/users

Syntax: `g_ssl_try_from` string

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossl](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_require_web](#), [g_ssl_try_out](#), [g_ssl_try_not](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_ssl_try_not - Skip ssl for these hosts

If the hosts match then SurgeMail Does not try ssl even if g_ssl_try_out matches.

Syntax: g_ssl_try_not string

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossll](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_require_web](#), [g_ssl_try_out](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_ssl_try_out - Try and start ssl mode to these hosts, may cause failures!

If the hosts match then SurgeMail tries to start SSL security on the SMTP session. Note that this may cause failures if the link is dropped by the receiving server.

Syntax: g_ssl_try_out string

See also: [ssl_pop_domain](#), [ssl_allow](#), [g_encrypt_ssl_force](#), [g_encrypt_ssl_noforce](#), [g_mirror_nossll](#), [g_ssl_allow](#), [g_ssl_allow_imap](#), [g_ssl_require](#), [g_ssl_require_imap](#), [g_ssl_require_login](#), [g_ssl_require_out](#), [g_ssl_require_web](#), [g_ssl_try_not](#), [g_ssl_try_from](#), [g_ssl_per_domain](#), [g_ssl_ciphers](#), [g_ssl_disable_sslv2](#), [g_ssl_sha1_sign](#)

g_stack - For testing only, NEVER SET THIS

Never set this, it can make the server unstable

Syntax: g_stack int

g_startup_delay - Startup delay

Seconds to wait before accepting inbound connections when starting SurgeMail .

Syntax: g_startup_delay int

g_store_dropped - Store upto 5000 bad bounces in the dropped directory

This is useful to check if vanish_bad_bounces is working correctly

Syntax: g_store_dropped bool

g_surbl - SURBL Spam URI Realtime Blocklists

This looks up each url found in each mail message and checks it against the SURBL database of your choice, the multi database can be used. See <http://www.surbl.org/>, adds headers of the form: X-Surbl: stamp urlfound nameofsurbl.

Syntax: g_surbl name=string stamp=string

Example: g_surbl name="multi.surbl.org"

stamp="sc.surbl.org,ws.surbl.org,phishing,ob.surbl.org,ab.surbl.org,jp"

See also: [g_honeypot_rbl](#), [g_myrbt_testing](#), [g_myrbt_to](#), [g_myrbt_store](#), [g_spam_allow_rbl](#), [g_surbl_reject](#), [g_surbl_whois](#), [g_surbl_skip](#), [g_surbl_skip_ip](#)

g_surbl_reject - Reject email with SURBL hits

This can reduce spam on your server by completely rejecting all email containing surbl web links...

Syntax: g_surbl_reject bool

See also: [g_honeypot_rbl](#), [g_myrbt_testing](#), [g_myrbt_to](#), [g_myrbt_store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl_whois](#), [g_surbl_skip](#), [g_surbl_skip_ip](#)

g_surbl_skip - URL's to allow even if listed in surbl

Sometimes you will want to whitelist a url that is listed in one or more surbl databases, use this setting to do that.

Syntax: `g_surbl_skip` string

See also: [g_honeypot_rbl](#), [g_myrrbl_testing](#), [g_myrrbl_to](#), [g_myrrbl_store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl_reject](#), [g_surbl_whois](#), [g_surbl_skip_ip](#)

g_surbl_skip_ip - Skip SURBL check if sender is from listed ip

Sometimes you will want to whitelist an ip from SURBL checks. Use this setting to do this.

Syntax: `g_surbl_skip_ip` string

See also: [g_honeypot_rbl](#), [g_myrrbl_testing](#), [g_myrrbl_to](#), [g_myrrbl_store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl_reject](#), [g_surbl_whois](#), [g_surbl_skip](#)

g_surbl_whois - Also check whois info on suspect urls - not for busy servers!

This setting searches whois information and compares what it finds to a list of known persistent spammers who register new domains regularly - if a match is found a surbl header is added. The whois servers don't like getting heavy load so don't use this setting if your server is very busy. A cache is used to minimize the load.

Syntax: `g_surbl_whois` bool

See also: [g_honeypot_rbl](#), [g_myrrbl_testing](#), [g_myrrbl_to](#), [g_myrrbl_store](#), [g_spam_allow_rbl](#), [g_surbl](#), [g_surbl_reject](#), [g_surbl_skip](#), [g_surbl_skip_ip](#)

g_surgeblog - Specialize SurgeMail as a Blog server

This setting causes SurgeMail's interface to specialize itself for the purposes of being a Blog server.

Syntax: `g_surgeblog` bool

g_surgeplus_delay_tell_upgrade - Delay informing existing users about new SurgePlus versions for

Delay informing existing users about new versions of SurgePlus for this long after the new version is downloaded to your server. SurgePlus clients poll the server once an hour so they won't be informed about the new version for up to an hour longer than the value of this setting. Use this setting combined with the `g_surgeplus_delay_tell_upgrade_exempt` setting so that only administrator users are informed about new versions at first so you can confirm the new version works fine with your existing server configuration before everyone upgrades. Example values: "3 hours" or "2 days"

Syntax: `g_surgeplus_delay_tell_upgrade` string

See also: [g_disable_surgeplus](#), [g_disable_surgeplus_updates](#), [g_surgeplus_delay_tell_upgrade_exempt](#)

g_surgeplus_delay_tell_upgrade_exempt - Users exempt from delayed new version informing

See the above setting for information. Example value: "user1@domain.name,user2@domain.name"

Syntax: `g_surgeplus_delay_tell_upgrade_exempt` string

See also: [g_disable_surgeplus](#), [g_surgeplus_delay_tell_upgrade](#)

g_surgeplus_hide_client_downloads - Hide the links to download and install SurgePlus Windows client

Use this setting if you don't want your users to know about the SurgePlus Windows client. All this setting does is to hide the download links from the web interface.

Syntax: `g_surgeplus_hide_client_downloads` bool

See also: [g_disable_surgeplus](#)

g_surgeplus_links - Add web links to SurgePlus from other web interfaces (and vice versa) for users allowed to use SurgePlus.

This causes links to appear in the SurgePlus interface to switch to using WebMail (and DBabble if you have the `g_dbabble_links` setting on).

Syntax: `g_surgeplus_links` bool

See also: [g_dbabble_links](#), [g_disable_surgeplus](#)

g_surgeplus_log_level - SurgePlus log level. 'none', 'info', or 'debug'. Default is 'info'

Sets the amount of logging done for SurgePlus. When using 'debug' level, data is logged to surgeplused.log in addition to surgeplus.log

Syntax: `g_surgeplus_log_level` string

Example: debug

See also: [g_log_level](#), [g_disable_surgeplus](#)

g_surgeplus_online - Enable online tracking in surgeplus
Not recommended.

Syntax: `g_surgeplus_online` bool

g_surgeplus_pop_server_name - Default pop server to set SurgePlus client download to connect to. SurgePlus Windows client downloads are set to connect to this POP server by default. This setting only applies if the user is downloading the client from a URL that does not match a valid domain on the server. If the URL does match a domain on the server, the domain specific version of this setting applies instead.

Syntax: `g_surgeplus_pop_server_name` string

See also: [surgeplus_pop_server_name](#), [surgeplus_smtp_server_name](#), [g_surgeplus_smtp_server_name](#)

g_surgeplus_port, g_surgeplus_secure_port - SurgePlus port and SurgePlus secure port.

SurgePlus uses the POP protocol to communicate with SurgeMail. However, some virus scanners running on the clients machine prevent the SurgePlus client from using POP commands that the virus scanner does not know about. In order to avoid this problem, SurgePlus uses port 7110 by default instead of port 110. However, clients not using a virus scanner (or clients using some virus scanners we have made SurgePlus work with - e.g. Norton) can safely use port 110 if they would otherwise be prevented from connecting to SurgeMail by a firewall. The SurgePlus client will quietly switch to using port 110 if it is not able to connect to the server using port 7110.

Syntax: `g_surgeplus_secure_port` int

See also: [old_xfile](#), [xfile_url](#), [g_pop_secure_port](#), [g_xfile_allow](#), [g_disable_surgeplus](#), [g_surgeplus_port](#)

g_surgeplus_smtp_server_name - Default smtp server to set SurgePlus client download to connect to. SurgePlus Windows client downloads are set to connect to this SMTP server by default. This setting only applies if the user is downloading the client from a URL that does not match a valid domain on the server. If the URL does match a domain on the server, the domain specific version of this setting applies instead.

Syntax: `g_surgeplus_smtp_server_name` string

See also: [surgeplus_pop_server_name](#), [surgeplus_smtp_server_name](#), [g_surgeplus_pop_server_name](#)

g_surgeplus_web_port - SurgePlus web port.

If you want your SurgePlus users to view shared files over a different port than WebMail uses give this setting a value.

Syntax: `g_surgeplus_web_port` int

See also: [xfile_url](#), [g_webmail_port](#), [g_disable_surgeplus](#), [g_surgeplus_web_url](#)

g_surgeplus_web_url - Direct SurgePlus users to access shared files at this url

Use this to override the default location that users are directed to to view shared SurgePlus web files. If you don't specify a value for this setting then it defaults to using the non-secure webmail port.

Syntax: `g_surgeplus_web_url` string

Example: `https://[domain]:7443`

See also: [xfile_url](#), [g_disable_surgeplus](#), [g_surgeplus_web_port](#)

g_surgewall_split - Split up surgewall messages, one per recipient
Split up incoming messages so subject tagging should work

Syntax: `g_surgewall_split` bool

g_surgeweb_backend_server - Backend machine to connect to
This specifies the backend machine where Surgeweb connects for email and to store user settings. Surgeweb will cache data here but store the master copy of anything on the backend machine.

Syntax: `g_surgeweb_backend_server` string

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#), [surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#), [g_surgeweb_restrict](#), [g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_benchmark - Log web request timing info for surgeweb benchmarking - matches ip addresses
Netwin testing use only

Syntax: `g_surgeweb_benchmark` string

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#), [surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#), [g_surgeweb_backend_server](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#), [g_surgeweb_restrict](#), [g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_debug - Log surgeweb debug info - matches ip addresses or email addresses - avoid
Note this setting should be used minimally as it affects performance

Syntax: `g_surgeweb_debug` string

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#), [surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#), [g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_logall](#), [g_surgeweb_restrict](#), [g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_disable - Disable access to SurgeWeb
Completely disable surgeweb access for whatever reason.

Syntax: `g_surgeweb_disable` bool

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#), [surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_work](#), [g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#), [g_surgeweb_restrict](#), [g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_idle_timeout - Idle timeout for surgeweb sessions (hours, default=48)
If no manual action is taken during this time the surgeweb session gets logged out

Syntax: `g_surgeweb_idle_timeout` int

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#), [surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#),

[g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#),
[g_surgeweb_restrict](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_logall - For requests matching g_surgeweb_debug also leave all webio & temp files - avoid Netwin testing use only

Syntax: g_surgeweb_logall bool

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#),
[surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#),
[g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_restrict](#),
[g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_remember_timeout - "Remember" timeout / max session length for surgeweb sessions (days, default=14)

Maximum time for Remember me and for single sessions

Syntax: g_surgeweb_remember_timeout int

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#),
[surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#),
[g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#),
[g_surgeweb_restrict](#), [g_surgeweb_idle_timeout](#)

g_surgeweb_restrict - Restrict surgeweb use to these accounts only
Allow surgeweb access to a matching set of email addresses

Syntax: g_surgeweb_restrict string

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#),
[surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_work](#),
[g_surgeweb_backend_server](#), [g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#),
[g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_surgeweb_work - Path to Surgeweb cache/work files

This is where Surgeweb stores it's temporary or working files, default I_G_HOME\surgeweb\work

Syntax: g_surgeweb_work string

See also: [surgeweb_backend_server](#), [surgeweb_backend_smtp](#), [surgeweb_backend_web](#),
[surgeweb_custom](#), [g_encrypt_surgeweb_show](#), [g_surgeweb_disable](#), [g_surgeweb_backend_server](#),
[g_surgeweb_benchmark](#), [g_surgeweb_debug](#), [g_surgeweb_logall](#), [g_surgeweb_restrict](#),
[g_surgeweb_idle_timeout](#), [g_surgeweb_remember_timeout](#)

g_tarpit_badrcpt - Delay rejection of bad recipients

Delay rejection of bad recipients (in seconds, default 4s).

Syntax: g_tarpit_badrcpt int

g_tarpit_blackhole - Reject email one recipient at a time to make spammers go away

If tarpit_blackhole is true then if it was going to drop the connection to that user. Instead it will keep it and let the user talk and try and send messages, but will reject all recipients, it only does this for a max of 200 channels, any more are dropped.

Syntax: g_tarpit_blackhole bool

g_tarpit_drop - Max recipients per hour from one IP

Drop link and ban for 1 hour if g_tarpit_max or g_max_bad_to has been exceeded.

Syntax: `g_tarpit_drop` bool

g_tarpit_max - Max number of local recipients per hour from one IP

If this limit is exceeded, the offending client is "tarpitted". This means the mail server starts pretending to go slowly. This is better than simply closing the connection as that will not stop the sending system from trying to reconnect rapidly or send to other systems rapidly, but tarpitting jams the sending system and limits the damage they can do to you and others. Cool huh?

Unlike `G_BOMB_MAX`, the `g_tarpit_max` setting counts the total of all recipients to all addresses from this IP address.

A setting of about 200-10,000 is probably good but be careful with mailing lists it will break them. Use an exclusion for IP addresses of known mailing lists or set the limit higher than known mailing lists, eg: 2,000 is probably a good setting just to avoid disasters without disrupting many real users.

Use `spam_allow ip.address.list` to over-ride the limit for known systems (eg: mailing list servers) that would be exceed the limit.

Syntax: `g_tarpit_max` int

g_tarpit_max_remote - Max remote recipients from one IP

The maximum number of remote recipients before slowing down.

Syntax: `g_tarpit_max_remote` int

g_tarpit_retry - Send retry error, 450 if tarpit limits exceeded

This setting has no further documentation currently available

Syntax: `g_tarpit_retry` bool

g_tcp_que_len - Length of listen queue for incoming connections

Default is 25 or 200 on windows, to reduce non paged pool on windows reduce to 20

Syntax: `g_tcp_que_len` int

g_tcp_read_timeout - Timeout in 'seconds' on POP connections (do not adjust)

Timeout in 'seconds' on POP connections, do not adjust. (default 600).

Syntax: `g_tcp_read_timeout` int

g_tellmail_ip - Tellmail IP restriction

Restrict remote tellmail commands to these IP addresses.

Syntax: `g_tellmail_ip` string

g_thread_max - Total maximum number of threads allowed

Total maximum number of threads allowed on this system. This should not normally be changed. If you do increase it start small, eg: 400 is a safe number on most systems. Generally if you need to increase it more than that then you have a performance problem that needs fixing and increasing it more is unlikely to be a good idea. On Linux if your `thread_max` setting is above 500 then you must modify `surgemail_start.sh` to increase the handle limit from 1024 to 2048 (at least twice the `g_thread_max` value). If you get crashes with 'handle_limit' recorded in the logs then it's likely that your operating system handle limit is too small for your `g_thread_max` setting. On Solaris you will need the 64 bit build of SurgeMail to increase this limit as the Solaris 32 bit 'c' libraries are limited to 256 file handles (I kid you not :-)

[See FAQ section on session limits](#)

Syntax: `g_thread_max` int

g_thread_reuse_real2 - Thread reuse

If enabled the server will reuse existing threads instead of creating and destroying threads for each incoming/outgoing message. This has no affect on performance but does avoid a bug in some UNIX threading libraries which leak handles and cause problems if threads are not reused. Generally best disabled except on early Linux systems.

Syntax: `g_thread_reuse2` bool

g_thread_spinlock - Spin more before sleeping when waiting for mutex

This setting has no further documentation currently available

Syntax: `g_thread_spinlock` bool

g_timeout_try_later - If timeout while waiting for message to arrive tell other end to retry

This 'may' cause faulty servers to endlessly retry a message. But should be ok. Normally this sort of timeout is very rare but can be caused by faulty virus scanner so retrying won't always help

Syntax: `g_timeout_try_later` bool

g_timezone - Timezone text

Text to be placed in the timezone part of the date string. e.g. +1200 NZT

Syntax: `g_timezone` string

g_timezone_force - Hours offset to local time, e.g. 5 (best left blank)

This setting has no further documentation currently available

Syntax: `g_timezone_force` int

g_to_valid - Require an @ and dotted domain in all dest addresses

This forces all destination addresses to contain a domain name (breaks cron job emails on unix)

Syntax: `g_to_valid` bool

g_tohost_local - Tohost entries to deliver locally

Authentication database tohost name entry to deliver locally. This setting only applies if `g_proxy` or `g_route_by_tohost` is enabled. This is useful to allow the configuration of multisite systems using `g_route_tohost` with a single shared authentication database.

Syntax: `g_tohost_local` string

g_toscan_path - Path used for mime parts for virus scanner

The default is the toscan directory under the home path, using this setting can help sometimes if permissions are a problem

Syntax: `g_toscan_path` string

g_uidl_big - Use random uidl if uidl not found

This can avoid uid collisions if uidl files are lost mysteriously

Syntax: `g_uidl_big` bool

g_unique_name - A unique name for this server

This name is used in place of the machine hostname in message filenames and thus friends confirmation message subjects

Syntax: `g_unique_name` string

g_url_alias - Allows translation from one URL to another

Allows translation from one URL or beginning of a URL to another. eg:

`g_url_alias` from="/cgi-bin/" to="/scripts/"

will cause the URL `http://localhost:7025/cgi-bin/fred.cgi` to reference the same file as `http://localhost:7025/scripts/fred.cgi` would have, the fred.cgi in the SurgeMail 'scripts' directory. The domain [url_alias](#) settings are checked before these, the first matching rule

is used, settings are checked in the order specified.

Syntax: g_url_alias from=string to=string ports=string

g_url_enable - Enables widearea url database

Syntax: g_url_enable <true/false>

If set then SurgeMail fetches the url database and updates from netwinsite.com every few hours. Messages which contain matches will get a header X-SpamUrl:... which will be used in the spam score. Once enabled you will contribute to Netwin's central server and also download from their once every couple of days.

Additions to your isspam/notspam training addresses are also sent to netwinsite.com (just the url's for white list/blacklist)

Syntax: g_url_enable bool

g_url_host_noscan - Disable the scan for url_host settings matching the domain in an incoming web request

SurgeMail uses g_server_name and url_host settings to determine the default domain to select for web requests, this setting stops it using the url_host settings (which may be slow on systems with a large number of domains)

Syntax: g_url_host_noscan bool

g_url_master - Not for general use

Used by netwin to manage the master server. Sorry this doesn't allow you to run your own master. Should be left blank

Syntax: g_url_master bool

g_url_master_to - Not for general use

Not for general use. Used by netwin for testing.

Syntax: g_url_master_to string

g_user_access - Allow / Restrict user access to features based on [g_access_group](#)

g_user_access group="wildcard" access="list"

This setting matches the [g_access_group](#) the user is in to the wildcard specified and applies the specified list to that user, giving / restricting thier access to certain features. The list may include any of the following:

Value	Result
alias	Access to the "Alias" page and features.
blog	Access to the "Blogs" page and features.
centipaid	Access to the "Centipaid" page and features.
delete	Access to the "Delete" button, which deletes the email account.
enotify	Access to the "Email Notification" page and features.
exceptions	Access to the "Exceptions" page.
filter	Access to filtering of messages. (g_filter_pipe , g_mfilter_file , g_dmail_filter)
friends	Access to the "Friends" pages, and system.
fwd	Access to the "Forwarding" features, forwarding, auto-responder.
fwdonly	Access to the "Forwarding" features. Without this only the auto responder is shown on the forwarding page
lists	Access to the "Lists" page and features.

log	Access to the "Log" page.
mailbox	Access to the "Mailbox" page, view mailbox, setup rules.
main	Access to the "Main" page containing user details.
pass	Access to the "Password" features, change password, password retrieval.
sms	Access to the "Sms" page.
spam	Access to the "Spam" page, and SmiteSpam and Asпам processing of messages.
spampriv	Access to the "Spam" pages' spam private feature
spf	Access to the "Spf" page and features.
surgeplus	Able to connect to SurgeMail using the SurgePlus client.
virus	Access to virus scanning of messages. (g_virus_cmd , g_virus_filter , g_virus_avast , g_scan_cmd)
webmail	Access to the "WebMail" button which logs the user into WebMail.

In addition you can prefix any of the above with ! to deny access. There are two other special case values, "all" and "none" which mean exactly what they say, access to "all" or "none" of the features.

Example:

```
g_user_access group="simple" access="all,!spam,!virus"
```

The above setting gives users in the 'simple' group access to all the features except spam and virus features.

Syntax: g_user_access group=string access=string

g_user_access_default - Default user features granted to users

This setting is a default access list for all users on the server, it is specified in the same maner as the [g_user_access](#) settings 'access' parameter. eg:

```
g_user_access_default "all,!spam,!virus"
```

Syntax: g_user_access_default string

g_user_access_from - When sending use from for useraccess rules

When sending a message the user access rules which are applied can be based on the 'from' header, this is not secure but is sometimes useful.

Syntax: g_user_access_from bool

See also: [smtp_auth_off](#), [surgewall_auth](#), [g_acctlog_authonly](#), [g_allow_user_authent_field_get](#), [g_allow_user_authent_field_set](#), [g_authent_always](#), [g_authent_any](#), [g_authent_allow_badascii](#), [g_authent_prefix_sep](#), [g_authent_process](#), [g_authent_cachelife](#), [g_authent_cachebad](#), [g_authent_cachesize](#), [g_authent_domain](#), [g_authent_encrypt_key](#), [g_authent_number](#), [g_authent_info](#), [g_authent_info_grp](#), [g_authent_ip](#), [g_authent_path_broken](#), [g_authent_single](#), [g_authent_spaces](#), [g_authent_strip_domain](#), [g_authent_restart](#), [g_authent_logall](#), [g_authent_fwdfile](#), [g_authent_timeout](#), [g_authent_last_login](#), [g_auth_hide](#), [g_auth_norelay](#), [g_auth_skipgateway](#), [g_mirror_nwauth](#), [g_mirror_nwauth_always](#), [g_filter_pipe_nwauth](#), [g_gateway_auth](#), [g_smite_skip_auth](#), [g_smtp_auth_debug](#), [g_smtp_portauth](#), [g_smtp_etrn_auth](#), [g_smtp_auth_off](#), [g_smtp_auth_ip](#), [g_smtp_nwauth](#), [g_smtp_nwauthm](#), [g_smtp_nwauth_msg](#), [g_spam_check_auth](#), [g_xauthuser_hide](#)

g_user_alias - Number of aliases accounts can create

This setting specifies the maximum number of account aliases an account (optionally in specified group) can create. The format of these aliases is specified in the file specified by the [g_user_alias_file](#) setting. eg.

```
g_user_alias quota="10" group=""
g_user_alias quota="20" group="grp1"
```

```
g_user_alias quota="30" group="grp2"

Syntax: g_user_alias group=string quota=int
```

g_user_alias_file - User aliases configuration file

This setting specifies the configuration file for user aliases. This file is in the following format:

domain alias_domain,access[,access]...

where domain is the domain name eg: email.com, alias_domain is the domain in which aliases can be created, and access specifies who is allowed to create these aliases, it can have one of the following values:

user	Users can create these aliases.
domadmin	Domain administrators can create these aliases.
admin	The Administrator can create these aliases.
private	Same as domadmin,admin. The Administrator and the Domain administrators can create these aliases.
public	Same as user,domadmin,admin. Everyone can create these aliases.

Example alias.dat file:

```
email.com *.email.com,public
email.com sport.email.com,public
internal.email.com email.com,private
internal.email.com internal.email.com,admin
```

```
Syntax: g_user_alias_file string
```

g_user_blogs - Number of blogs accounts can create
Specifies blog limit based on user group.

```
Syntax: g_user_blogs group=string quota=int
```

Example: g_user_blogs group=premium quota=15

See also: [blogs_max_per_user](#), [url_blogs](#), [g_access_group](#), [g_blogs_enable](#), [g_blogs_maximum_image_width](#), [g_blogs_maximum_image_size](#), [g_blogs_maximum_items_in_top_page](#), [g_blogs_max_per_user](#), [g_blogs_default_template](#), [g_blogs_use_sub_domains](#), [g_blogs_sub_domain_prefix](#), [g_blogs_not_unique](#), [g_blogs_not_global](#), [g_blogs_no_suffix](#), [g_blogs_ping](#), [g_blogs_domonly](#), [g_blogs_image_optional](#), [g_blogs_allow_links](#), [g_blogs_cleanup_links](#), [g_blogs_comment_rev](#)

g_user_cookies - Enable browser cookies for user self management

Enable browser cookies for user self management.

```
Syntax: g_user_cookies bool
```

g_user_delete - Let users delete themselves
Enables the user delete button in the user self management page, assuming the use access rules also allow it

```
Syntax: g_user_delete bool
```

See also: [g_aspam_headers](#), [g_aspam_need_ip](#)

g_user_domainlist - Show domains list on user pages

This setting decides who will see the drop-down list of domains on the user check, add, login, and management pages. It has three possible values: user, domadmin and admin. A value of 'user' allows everyone to see the list, 'domadmin' allows domain admins and the admin to see the list, and 'admin' allows only the admin to see the domains list.

Syntax: `g_user_domainlist` string

g_user_filter_early - Process user exceptions/filters before tagging message as spam

Causes the users exception rules to be processed before tagging the message as spam, meaning, if a rule matches to 'accept' a message, that message not to be tagged as spam.

Syntax: `g_user_filter_early` bool

g_user_friends_domain_log_disable - Disable domain level friend.log file

By default a friend.log file is written to each domain mailbox_path. This file is a collection of all users friends.log entries that rotates when it reaches 2mb in size.

Syntax: `g_user_friends_domain_log_disable` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_log_disable](#)

g_user_friends_log_disable - Disable user level friend.log file

By default a friend.log file and 1 rotation is written for each user. Each log should only be approx 10k in size.

Syntax: `g_user_friends_log_disable` bool

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#)

g_user_list_quota - Number of mailing lists users can create

`g_user_list_quota` group="" quota="100"

This setting configures the number of mailing lists a user can create on this server. The group field is optional, specifying none effects all users globally, otherwise it matches this against the users access group. See also `user_list_quota` which can set quota per domain. Also the `list_quota` authent field can set quota per user.

Syntax: `g_user_list_quota` group=string quota=int

g_user_mail_view - Whether an admin can view/display users inbox mail

This setting enables the 'view' links on the users mailbox page. These links will show the content of the users email. They also log the access to the users log file, identifying the IP from which the admin viewed the message.

Syntax: `g_user_mail_view` bool

g_user_mfilter - Local delivery Mfilter rules

Mfilter rules to run late in the delivery process after the email messages have become "user specific", In particular this allows filtering based on the output of g_user_pipe.

Syntax: g_user_mfilter string

g_user_pipe - Local delivery filter pipe

Pipe run on file just before delivery to user, \$USER\$ available on command line. This allows the message to be modified (also see g_filter_pipe).

Syntax: g_user_pipe string

g_user_receive_rule - Define valid source addresses for users in a group

This setting has no further documentation currently available

Syntax: g_user_receive_rule group=string from=string

g_user_send_ip - Block any ip address from sending too many emails

This does not apply to g_user_send_white addresses. This will also enable counting of sends for users using g_relay_window. Whitelist ip addresses with g_user_send_white setting. This limit is 'per day'

Syntax: g_user_send_ip int

g_user_send_rule - Define valid recipient addresses for users in a group (requires SMTP AUTH)

This rule allows you to define which domains users in the specified group can send email to.

g_user_send_rule group="wildcard" to="number"

If 'group' is set to '*' then it applies to users who are not in a group (see [g_access_group](#)), and/or whose group does not match another g_user_send_rule setting. The 'to' field contains a wildcard list of allowed email addresses.

Syntax: g_user_send_max group=string max=int

g_user_send_rule - Define valid recipient addresses for users in a group (requires SMTP AUTH)

This setting has no further documentation currently available

Syntax: g_user_send_rule group=string to=string

g_user_send_warning - Warn manager if any user sends more than this many messages per day, e.g. 5000

This setting is useful to detect a spammer sending out bulk email from your system, this setting only applies to authenticated users, so someone who has figured out the password of one of your users (or a virus on their computer) or a registered user of some sort. If g_user_send_ip is defined then warnings will also be sent if an ip address exceeds this limit.

Syntax: g_user_send_warning int

g_user_send_white - No limit for these ip addresses/users

This is a white list for the ip and user send limits.

Syntax: g_user_send_white string

g_user_sms_quota - SMS quota

Number of SMS messages accounts can send.

Syntax: g_user_sms_quota group=string initial=int period=string

g_user_status_send - Number of days after which to send user status messages (0 = never)

When the user enables friends then this setting will send them a regular report on what is pending and what filter rules have done.

Syntax: g_user_status_send int

See also: [friends_at_rcpt](#), [friends_pending_name](#), [g_friends_only](#), [g_friends_bounce_rej](#), [g_friends_name](#), [g_friends_pending_name](#), [g_friends_silent](#), [g_friends_ignore](#), [g_friends_skip_ip](#), [g_friends_confirm_subject](#), [g_friends_default_mode](#), [g_friends_default_autoadd](#), [g_friends_latest_headers](#), [g_friends_pending_keep](#), [g_friends_pending_vanish](#), [g_friends_at_rcpt](#), [g_friends_allow_spf](#), [g_friends_spf_fail_bounce](#), [g_friends_check_spf](#), [g_friends_byweb](#), [g_friends_always](#), [g_friends_add_trusted](#), [g_friends_confirm_debug](#), [g_friends_rotate](#), [g_friends_short](#), [g_friends_ignore_trusted](#), [g_friends_url](#), [g_friends_bounce_second](#), [g_friends_old_status_email](#), [g_friends_spam_score](#), [g_friends_release_wash](#), [g_imap_friends](#), [g_quota_friends](#), [g_user_friends_domain_log_disable](#), [g_user_friends_log_disable](#)

g_user_utoken_days - Length of time a user self management login token is valid

Length of time a user self management login token is valid for. Length of time a user self management cookie is valid for. After this time period the login token will stop allowing the user access and they will need to login again.

Syntax: g_user_utoken_days int

g_user_utoken_expire - Length of time a user self management login token is valid for

This setting has no further documentation currently available

Syntax: g_user_utoken_expire int

g_user_utoken_idle - Length of time a user self management login token may remain idle for

This setting has no further documentation currently available

Syntax: g_user_utoken_idle int

g_user_virus_scan - Allow users to enable / disable virus scanner for themselves

This setting adds a tickbox to the Spam page in user self administration that allows the user to enable and disable the virus scanner for them selves.

Syntax: g_user_virus_scan bool

g_vanish_any_bounce - Vanish all bounces that are not bounces to messages from this machine (requires g_received_name)

This setting will vanish spam pretending to be a bounce, it is possible it will vanish a real but badly formed bounce (badly formed as it contains no indication that it came from this server).

Syntax: g_vanish_any_bounce bool

g_vanish_bad_bounces - Vanish suspected spam bounces

Vanish suspected spam bounces (requires g_received_name).

Syntax: g_vanish_bad_bounces bool

g_vanish_virus_bounces - Vanish suspected virus bounces (requires g_received_name)

This setting gets rid of most of those stupid virus bounces you get from emails you haven't sent. It works by checking incoming virus bounces for the received header that must exist if it was sent with your mail server. If the header is not found, the message is dropped. Recomendend.

Syntax: g_vanish_virus_bounces bool

g_verify_helo - Verify helo name translates to same network as sending system.

Syntax: g_verify_helo "true/false"

It will skip this check for any trusted connection (smtp authenticated, or any ip it would allow to forward)

It adds this header:

X-Verify-Helo

It simply takes the helo name, and turns it into a number a.b.c.d, then it checks that the connection is coming from 'a.b.*.*' if it isn't it adds a header saying as much.

Syntax: g_verify_helo bool

See also: [g_spam_grey_verify](#), [g_verify_smtp](#), [g_verify_timeout](#), [g_verify_mx](#), [g_verify_mx_skip](#), [g_verify_image_hard](#)

g_verify_image_hard - Use extra difficult human verification image (used in blogs)

This setting has no further documentation currently available

Syntax: g_verify_image_hard bool

g_verify_mx - Verify sender IP by MX

Verify MX records contain senders IP address (also see g_verify_mx_skip).

Syntax: g_verify_mx bool

See also: [g_spam_grey_verify](#), [g_verify_smtp](#), [g_verify_timeout](#), [g_verify_mx_skip](#), [g_verify_helo](#), [g_verify_image_hard](#)

g_verify_mx_skip - Skip verify sender IP by MX

Use to define incoming mail gateway IPs so the MX verify doesn't fail on them.

Syntax: g_verify_mx_skip string

See also: [g_spam_grey_verify](#), [g_verify_smtp](#), [g_verify_timeout](#), [g_verify_mx](#), [g_verify_helo](#), [g_verify_image_hard](#)

g_verify_smtp - Verify SMTP port

Verify we can talk back to the SMTP port on incoming IP address.

Syntax: g_verify_smtp bool

See also: [g_spam_grey_verify](#), [g_verify_timeout](#), [g_verify_mx](#), [g_verify_mx_skip](#), [g_verify_helo](#), [g_verify_image_hard](#)

g_verify_timeout - Seconds to wait for SMTP response, default is 10 seconds

As the verification of incoming addresses is done while the message is arriving at the 'data' stage, it is critical that it not take more than 30-60 seconds or the sending server will give up and the message will be lost. Generally this setting should not be changed.

Syntax: g_verify_timeout int

See also: [g_spam_grey_verify](#), [g_verify_smtp](#), [g_verify_mx](#), [g_verify_mx_skip](#), [g_verify_helo](#), [g_verify_image_hard](#)

g_virus_allow_unmonitorable - Allow unmonitorable content (avast antivirus)

By default messages that cannot be scanned (eg as they contain password protected archive files) are blocked by the avast virus scanner. This setting allows unmonitorable content to be sent.

Syntax: g_virus_allow_unmonitorable bool

g_virus_avast - Enable Avast virus scanner integration

Enable Avast virus scanner integration. Avast should first be licensed and installed before this is enabled. Installation is done by pressing the install button next to this setting in the global settings page. Licensing is part of the SurgeMail key. During the SurgeMail evaluation period full Avast license is available. Subsequent to that the Avast integration must be purchased.

Status of the progress of installation and the whether Avast is currently uptodate is displayed in the main status page.

Syntax: g_virus_avast bool

g_virus_avast - Set Avast update time

This is a string based setting that allows you to specify when Avast updates are attempted.

eg: to update at 12 midnight, 6am,12noon and 6 pm.

g_virus_avast_hour "0,6,12,18"

Syntax: g_virus_avast_hour int

g_virus_cmd - Command line virus checker to run on MIME parts

If defined the mail server will extract MIME parts in a multi part message and run the virus scanner over the extracted file. The command line can include \$FILE\$ which will be replaced with the actual file name of the extracted part. An intelligent cache is used so mailing lists, etc, will not require running the virus scanner on every message sent. If you set this to "do_not_run" then SurgeMail will extract the MIME parts but not actually run any program, some virus scanners scan all files on the system so the file is deleted magically and SurgeMail will notice and bounce the message. If your scanner supports the returning of return codes if a virus is found then you should use g_virus_cmd_codes with this setting as this is more reliable than having to detect if a file is deleted and also means also will work on viruses in archives which a lot of scanners won't delete.

Syntax: g_virus_cmd string

g_virus_cmd_codes - Return codes to bounce message

Accept return codes from virus scanner as a confirmation that the scanned file is infected, eg: 1,2,3,4,5.

Lets SurgeMail check the return code from g_virus_cmd and if the code matches one in the above setting assumes its a virus and bounces it.

g_virus_cmd_codes "10,12"

This would assume its a virus if the scanner returns return code 10 or 12 and then will bounce the message.

Syntax: g_virus_cmd_codes string

g_virus_cmd_drop - Drop silently instead of reject at data stage - not recommended

This should only be used when your front end server is not scanning for viruses and your back end server then rejects the message generating back scatter on the front end server.

Syntax: g_virus_cmd_drop bool

g_virus_cmd_email - Set if scanner can understand email message files

If this is set then the scanner is responsible for extracting the mime parts of a message and scanning them

Syntax: g_virus_cmd_email bool

g_virus_cmd_max - Maximum number of concurrent threads to use for scanning

Syntax: g_virus_cmd_max "number of threads"

This sets the maximum number of threads that be used for running the virus scanner set by g_virus_cmd. Some scanners can take a while to scan a message and if the server is very busy this can tie up many channels and drain the cpu slowing down the entire mail server. When the maximum has been reached any messages coming in will be passed on without being run through the scanner - although this is not the best, it's better than the mail server grinding to a halt.

Syntax: g_virus_cmd_max int

g_virus_cmd_nodel - Do not delete scanned files

Disables cleanup of scanned files, so you can test manually. The files are extracted to the "toscan" directory inside the SurgeMail directory. You should never normally need this on unless for debugging purposes.

Syntax: g_virus_cmd_nodel bool

g_virus_cmd_size - Max size of messages to scan
Useful to stop scanning of huge files, e.g. 1mb or bigger

Syntax: g_virus_cmd_size int

g_virus_cmd_sleep - Wait after g_virus_cmd incase delete is not immediate

Milli seconds to wait after g_virus_cmd incase delete is not immediate, eg: 500 = half a second.

Syntax: g_virus_cmd_sleep int

g_virus_disable_local - Disable scanning for local trusted users
Skip virus scanner for authenticated users and 127.0.0.1

Syntax: g_virus_disable_local bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast_hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_disable_remote - Disable virus scans for non-local addresses

By default SurgeMail scans incoming messages from non-local senders, this disables that behaviour so scans will only occur if any recipient has virus scan access. You will probably need g_user_virus_scan true as well.

Syntax: g_virus_disable_remote bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast_hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_filter - Virus checker or filter that takes commands on stdin and response on stdout

Virus filters use the following protocol the process is run continuously and sent on STDIN a command of the form, "nnn CHECK fullfilename envelopefilename\\n" and in response it must send back is "nnn OK|REJECT|ERROR reason text\\n"

It can modify the file directly and then respond with 'ok', however if it does this it must maintain the crlf line terminated and dot stuffed nature of the file.

Here is an example test of a virus filter

```
c:\surgeemail> vfilter.exe
1 check c:\surgeemail\work\a.itm c:\surgeemail\work\a.hdr
1 REJECT Found something bad in that file
2 check c:\surgeemail\work\a.itm c:\surgeemail\work\a.hdr
2 OK send message along
```

a.hdr would contain:

```
From: bob@domain.com
To: xyz@thisdomain.com
To: xyz3@thisdomain.com
```

Syntax: g_virus_filter cmd=string type=string

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast_hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_filter_require - Require filter pipe

If any g_virus_filter pipe fails bounce messages rather than allow to continue.

Syntax: g_virus_filter_require bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_fprot - Set F-PROT port for mail scanning

Typically set this to 11200

First install f-prot virus scanner, exact steps will vary depending on platform so follow your F-Prot install instructions, but as an example on Linux we did this:

```
cd /usr/local
gunzip DISTRIBUTION.tar.gz
tar -xvf DISTRIBUTION.tar
cd f-prot
./install-f-prot.pl
cd tools

# Now start mail scanner as user 'mail'
su mail -c "/usr/local/f-prot/tools/scan-mail.pl -server -daemon"
```

Your will also need to start the scanner as above in your startup scripts (e.g. rc.local)

Then lastly in surgemail.ini set

g_virus_fprot 11200

When a message is scanned a header X-Fprot: ... is added giving some informational status.

Syntax: g_virus_fprot int

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_late - Run virus scan after most spam filter processing

This can reduce load on virus scanner which is often a slow process

Syntax: g_virus_late bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#)

g_virus_localhost - Don't skip virus checks for 127.0.0.1 originating emails

This setting should not normally be used, it will make it scan locally generated emails, dlist messages etc...

Syntax: g_virus_localhost bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#)

g_virus_recent_skip - Skip recent virus cache

Skip virus recent cache which attempts to speed up virus scanners.

Syntax: `g_virus_recent_skip` bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_rename - Rename attached executables to prevent autorun

If enabled SurgeMail will rename dangerous executable files by replacing the '.' with an '_'. This will stop many autorun viruses. This is name

Syntax: `g_virus_rename` bool

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_report](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_report - Report detected viruses to someone

Sends an email report to the specified address when a virus comes in.

Syntax: `g_virus_report` string

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_restart](#), [g_virus_late](#), [g_virus_localhost](#)

g_virus_restart - Restart vpipe virus scanners

Restart vpipe virus scanners every this many items.

Syntax: `g_virus_restart` int

See also: [g_user_virus_scan](#), [g_vanish_virus_bounces](#), [g_virus_avast](#), [g_virus_avast hour](#), [g_virus_allow_unmonitorable](#), [g_virus_cmd](#), [g_virus_cmd_codes](#), [g_virus_cmd_nodel](#), [g_virus_cmd_size](#), [g_virus_cmd_sleep](#), [g_virus_cmd_max](#), [g_virus_cmd_drop](#), [g_virus_cmd_email](#), [g_virus_disable_remote](#), [g_virus_disable_local](#), [g_virus_filter](#), [g_virus_filter_require](#), [g_virus_fprot](#), [g_virus_recent_skip](#), [g_virus_rename](#), [g_virus_report](#), [g_virus_late](#), [g_virus_localhost](#)

g_vpipe_concurrent - Concurrent requests to vpipe process

Concurrent requests to vpipe process, default is 7, set to 1 to debug vpipe issues

Syntax: `g_vpipe_concurrent` int

g_vpipe_notag - Disable vpipe result headers

Disable headers showing vpipe results in messages.

Syntax: `g_vpipe_notag` bool

g_vpipe_skip - Skip virus filter checks per IP address

Disable virus and crc checking for known safe bulk mailers that would otherwise overload the server. This setting affects the virus checker.

Example: `g_vpipe_skip "20.0.0.2"`

- `g_virus_cmd`
- virus filters (`g_virus_filter`)
- filter program (`g_filter_pipe`)

- F-Prot in daemon mode (g_virus_fprot)

Syntax: g_vpipe_skip string

g_vpipe_timeout - Timeout for virus filters (default 60s)

The timeout in second that SurgeMail will wait for a virus filter (defined by g_virus_filter) to complete. If after this time the virus filter has not responded the message will be let through and the following line logged in mail.log:

"Virus filter not responding, stuck on <msg file> allowing message through"

Syntax: g_vpipe_timeout int

g_web_access_grp - Restrict user groups to specific ports

Specifies a user group or groups and a list of valid web ports for that group.

Syntax: g_web_access_grp group=string ports=string

g_web_access_ip - Restrict access to web ports based on ip

Specifies a list of ports and a wildcard list of valid ip addresses who can connect to those ports.

Syntax: g_web_access_ip ports=string ip=string

g_web_access_max - Maximum number of concurrent web logins for group

Specifies the maximum number of concurrent web logins for a certain group of users.

Syntax: g_web_access_max group=string max=int

g_web_admin_max - Maximum number of concurrent web admin sessions

Web admin requests are recorded, the remote IP and local port are used to identify a particular session. This setting places a limit on the number of sessions at any one time.

Syntax: g_web_admin_max int

See also: [g_admin_utoken_expire](#), [g_admin_utoken_idle](#)

g_web_charset - Charset for html pages

Sets the charset to use for each language i.e. e.g. iso-8859-1

Syntax: g_web_charset lang=string charset=string

g_web_force_doctype_first_disable - Disable webserver behaviour to force doctype definitions to be displayed first.

Comments displayed on the webpages (including template filenames), mean IE does not use the doctype definition. Surgemail tries to display doctype first. This setting reverts to old behaviour.

Syntax: g_web_force_doctype_first_disable bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_url_path](#), [g_web_title](#)

g_web_hide_source_names - Hide the name of the source template page in output web pages.

To aid tailoring each web page in the web admin shows it's own address so you can find it to modify it. Some admins consider this a security issue, or just a bit ugly, so use this setting to hide this information when you don't need it.

Syntax: g_web_hide_source_names bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_web_max - Max concurrent web connections, default is 100

This includes web admin, webmail etc...., The default limit should be sufficient for most systems. Although a limit of 10 would be tons for most systems we had to set the default high as this setting was added recently.

Syntax: **g_web_max** int

g_web_max_perip - Max concurrent web connections per-ip, default is 30

This includes web admin, webmail etc...., The default limit should be sufficient for most systems unless all your users are coming through a common proxy

Syntax: **g_web_max_perip** int

g_web_noserver - Disable Server header in http responses

Some security firms require this in order to hide the software application information

Syntax: **g_web_noserver** bool

g_web_old_behaviour - Revert to old style webserver behaviour

To pass various auditing tests admin interface no longer responds to arbitrary url. This restores old behaviour.

Syntax: **g_web_old_behaviour** bool

g_web_ref_path_extension - Path extension to add to web page image/css references.

This setting is used for caching purposes. See [SurgeMail template caching](#) for details

Syntax: **g_web_ref_path_extension** string

g_web_timeout - Timeout for web requests

Timeout for web requests, the default is 180 seconds, generally it should not be set below 61 seconds

Syntax: **g_web_timeout** int

g_web_title - Title to use on specified web page

This lets you customize the title of each management web page.

Syntax: **g_web_title** page=string title=string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#)

g_web_url_path - Url to path translation with access specifier

This lets you set up aliases and translations of urls partly based on the access rights of the user.

Syntax: **g_web_url_path** url=string path=string access=string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#),

[g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_title](#)

g_webdav_enable - Enable webdav access for users

Enable 'webdav' features so users can store data, you must also define [g_webdav_path](#)

Syntax: [g_webdav_enable](#) bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webdav_group - Only allow webdav if member of webdav access group

Require that users be members of the webdav group

Syntax: [g_webdav_group](#) bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webdav_path - Root path for webdav storage

For example c:\surgemail\webdav

Syntax: [g_webdav_path](#) string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webdav_public - Enable non authenticated access to pub folder (readonly)

This setting enables the user to place web pages (static) up on their email account, the public url would be [http://your.server/wd/username/pub/...](#)

Syntax: [g_webdav_public](#) bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_limit - Maximum number of concurrent webmail requests

This should not generally be adjusted, it is simply a limit to prevent DOS attacks or overloading from web requests. A value of 10-300 would be reasonable. The default is 200

Syntax: [g_webmail_limit](#) int

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#),

[g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_popmode - Use POP3 instead of IMAP in WebMail.

This results in pophost being passed to webmails domain configuration file, surgehost.ini. If you change this setting you should delete surgehost.ini and run "tellmail surgehost_update" to rebuild it.

Syntax: g_webmail_popmode bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_port - WebMail port (default 7080)

This is the port that WebMail users should connect through (unless you want better security, then use the secure port and HTTPS protocol listed below) By default it is port 7080, but if you are not running a web server you probably want to change it or add port 80, eg:"7025,80" so that people can get to it with a URL like this: http://your.mail.server instead of http://your.mail.server:7080. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_webmail_port int

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_secret - Secret string used by webmail when sending the ip address of connecting users

This is used with webmail when you want surgemail access rules to apply to webmail users, webmail has a matching setting which makes it pass the ip address through

Syntax: g_webmail_secret string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_secure_port - WebMail secure port (default 7443)

This is the port that WebMail users should connect through.. By default it is port 7443, but if you are not running a web server you probably want to change it or add port 443, eg:"443" so that people can get to it with a URL like this: https://your.mail.sever Instead of https://your.mail.server:7443. Use the keyword 'disabled' to disable this part of the SurgeMail service.

Syntax: g_webmail_secure_port int

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_select_domain - Send select_domain instead of host in webmail autologins

Recommended. This uses the select_domain method of auto-logins with WebMail, it often works where the old method fails.

Syntax: `g_webmail_select_domain` bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_timeout - Timeout for webmail or any cgi process (in seconds, default 360)

If the webmail cgi fails to respond this limits how long SurgeMail will wait before killing the process.

Syntax: `g_webmail_timeout` int

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_url - Url to the WebMail cgi

If WebMail is not in the default place and/or is not on the SurgeMail machine then this setting tells SurgeMail where it is so links to WebMail from SurgeMail function correctly.

Syntax: `g_webmail_url` string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_urladd - Url data to append to WebMail auto-login link

This setting allows you to specify additional information and settings which are passed to WebMail when SurgeMail links to it.

Syntax: `g_webmail_urladd` string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_useip - Use the ip address in `g_webmail_port` setting

By default it will use the same url as the user connects on which is generally better.

Syntax: `g_webmail_useip` bool

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_workarea](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_webmail_workarea - Path to WebMail workarea

If WebMail is not installed in the default location on this SurgeMail machine this setting tells SurgeMail where to find it.

Syntax: g_webmail_workarea string

See also: [webmail_url](#), [webmail_urladd](#), [webmail_workarea](#), [webmail_host](#), [web_url_path](#), [g_keepalive](#), [g_key_manual](#), [g_key_nowarning](#), [g_webdav_enable](#), [g_webdav_group](#), [g_webdav_public](#), [g_webdav_path](#), [g_webmail_limit](#), [g_webmail_port](#), [g_webmail_secure_port](#), [g_webmail_timeout](#), [g_webmail_useip](#), [g_webmail_popmode](#), [g_webmail_url](#), [g_webmail_urladd](#), [g_webmail_select_domain](#), [g_webmail_secret](#), [g_web_hide_source_names](#), [g_web_force_doctype_first_disable](#), [g_web_url_path](#), [g_web_title](#)

g_work - Workarea Path

Work area for SurgeMail temporary work files.

Syntax: g_work string

g_xauthuser_hide - Hide X-Authenticated-User header

The header X-Authenticated-User is added to all local deliveries for users that login using SMTP authentication. This is the most reliable way to determine who actually sent this email. This setting will disable the addition of this header.

Syntax: g_xauthuser_hide bool

g_xfile_allow - IP address to allow xfile and WebMail features from

Allow xfile & web upload features for users. Set to '*' or the WebMail servers IP address.

Syntax: g_xfile_allow string

See also: [old_xfile](#), [xfile_url](#)

g_xrcpt_hide - Hide X-Rcpt header

The X-Rcpt header is added indicating which local account this message was delivered to. This setting will disable the addition of this header.

Syntax: g_xrcpt_hide bool

g_xrcptoriginal_hide - Hide X-Rcpt-Original header

The X-Rcpt header is added indicating which local account this message was delivered to. If the mail has been redirected for any reason the original delivery address is added as an X-Rcpt-Original header. This setting will disable the addition of this header.

Syntax: g_xrcptoriginal_hide bool

g_xserver_hide - Hide XServer header

This will hide the X-Server header.

spamlist - Spam Filter Rules

These rules allow simple filtering of Email messages for common or repetitive spam message. The form lets you specify whether a string is found in a specified header that all such messages be bounced or redirected. This form will write or modify your mfilter.rul file to include an auto generated section which obeys the rules you have defined, e.g.

```
D:\>type \surgemail\mfilter.rul
# BEGIN_AUTO Generated section do NOT EDIT this bit
if (isin("Subject","bad words")) accept "fred@remote.domain"
if (isin("To","bad words")) accept "fred@remote.domain"
# END_AUTO Generated section do NOT EDIT this bit
```

You can write much more complex rules yourself manually, see [mfilter.htm](#) for more details.

Syntax: g_xserver_hide bool

Compatibility settings

g_authent_prefix_sep - Authent Prefix Separator (deprecated - for backward compatibility only)

Prefix separator for prefix based separator. Only relevant if enabled on a per vdomain basis using the "prefix" setting.

g_authent_fwdfile - Use DMail forward files (deprecated - for backward compatibility only)

Allows old style DMail forward files to be read.

g_dmail_filter - Run DMail compatible filter files (deprecated - for backward compatibility only)

Run DMail compatible filter files. Mfilter rule files should be used instead.

g_mirror_nwauth - Mirror NWAuth data files (deprecated - for backward compatibility only)

This setting is no longer used (as of SurgeMail 1.7d), the g_mirror_mode setting is used instead to decide whether do mirror the NWAuth database.

Specialist / debugging settings

g_backtrace_disable - Backtrace Disable

Disable backtrace information for unix systems.

g_crash_normal - Crash without catching exceptions

Crash without catching signals 10,11. In particular this will generate correct core files on FreeBSD systems.

g_debug_block - For catching bugs in block file processsing

For catching bugs in block file processsing.

g_mutex_timeout - Crash without catching exceptions

Default mutex timeout period in seconds (default=600 ie 10minutes). This is a self monitoring feature that if it has not received a mutex for some reason (usually a bug, but could be server overloading) SurgeMail will shut itself down. If g_restart is enabled this would restart surgemail.

g_shutdown_slow - Delay shutdown

Add 20 second delay to shutdown for testing purposes only.

g_slow_welcome - Delay the welcome message

Add 20 second delay to welcome message for testing purposes only.

g_vpipe_fail_crash - Crash if vpipe fails

Crash SurgeMail if vpipe fails. This is for debugging purposes only.

WebMail settings

WebMail Manual

Click [here](#) to view the complete WebMail manual (online).

Autologin

Whenever a user clicks a link in WebMail that takes them to a SurgeMail feature like changing password, holiday settings, etc.. WebMail does what is called an 'auto-login'. There are two different auto-login methods, one uses a temporary file the other uses the POP server. In both cases the users password is encrypted, stored and then deleted all in the space of time it takes the page to load in the browser.

A common problem with the autologin is configuration. SurgeMail and WebMail need to be configured to both use the same method. The table below shows the settings required in each config file

Method	SurgeMail.ini	WebMail.ini
POP Server	g_autologin_pop "TRUE"	use_id_autologin true
Temporary file	g_autologin_pop "FALSE"	use_id_autologin false

You can find surgemail.ini in the Windows directory, or /etc (on UNIX). If you modify that file directly you MUST either restart the SurgeMail server or run 'tellmail reload'. WebMails config, webmail.ini, can be found in the SurgeMail scripts directory.

Configuring login for multiple domains

Configuring WebMail so that your users can move easily from the SurgeMail user configuration interface and WebMail can be tricky because there are so many different configuration options in both SurgeMail and WebMail. SurgeMail will now automatically add vhost settings to the WebMail configuration, using the surgehost.ini file in the surgemail/web_work directory. If you need to manually configure vhost settings, this should still be done in the webmail.ini file as always.

As a general rule you want to have a vhost section in your webmail.ini for every domain you have setup in SurgeMail beyond the first. So for example if you have these 3 domains setup in SurgeMail domain1.com, domain2.com and domain3.com you will have 2 vhost sections in webmail.ini like this...

```
pophost domain1.com
domain domain1.com

vhost domain2.com
pophost domain2.com
domain domain2.com
suffix @domain2.com

vhost domain3.com
pophost domain3.com
domain domain3.com
suffix @domain3.com

vend
```

If domain1.com, domain2.com and domain3.com all resolve to the same machine then the additional pophost settings may seem redundant BUT they are required to ensure that WebMail uses the same data files for any given user when they login manually and when they move from the SurgeMail user interface to WebMail.

Smart Router / Load Balancer

If you're using a load balancer with WebMail then please read [this](#).

Authentication Modules - a guide

- [List of modules](#)
- [Testing modules](#)
- [Choosing a module](#)
- [Configuring modules](#)
- [Fields used by SurgeMail](#)
- [Mixed case usernames and domains](#)
- [Example configuration of firstname and lastname for LDAPAuth](#)

SurgeMail supports external authentication modules which are simple command line based programs that understand a small set of commands to add, remove and look up user details in your user database.

We provide modules for most common databases, including:

- [NTAuth](#) - Windows user database - in surgemail distribution (Windows)
- [UnixAuth](#) - Unix password files - in surgemail distribution (UNIX)
- [NWAuth](#) - NetWin's own user database - in SurgeMail distribution
- [ODBCauth](#) - ODBC data sources, i.e. Microsoft SQL databases
- [MySQLAuth](#) - MySQL UNIX based SQL databases
- [LDAPAuth](#) - LDAP database
- [RadiusAuth](#) - Radius database
- [OracleAuth](#) - Oracle database
- [PAMAuth](#) - Linux PAM

We also have a few utilities for running the above modules in different ways, including:

- [MultiAuth](#) - Multiple modules simultaneously
- [TCPAuth](#) - TCP network connect

All of these modules can be found [here](#) along with instructions on how each can be configured.

Of course, you can also write your own [here](#) is the protocol definition.

Authent modules should always be tested at the command line to see if they are working. Here is an example using NWAuth, the standard NetWin module:

```
c:> nwauth
set bob@test.com bob
+OK bob@test.com added to database
lookup bob@test.com
+OK bob@test.com config 0
check bob@test.com xxx
-ERR bob@test.com password wrong or not a valid user
search bo*@test.com
+DATA bob@test.com
+DATA bobcat@test.com
+OK Search Complete 2 items found out of 1510
set bob@test.com bob quota="200" fwd="fred@test.com"
+OK bob@test.com added to database
lookup bob@test.com
+OK bob@test.com config 0 quota="200" fwd="fred@test.com"
```

Choosing an Authent Module.

The web admin GUI will list available authent modules and guide you to the config pages for each authent module. Most authent modules have an ini file that needs to be configured, eg: odbcauth.ini or ldapauth.ini and a related binary.

When you download an authent module all files should be placed in the SurgeMail directory.

Again, test the authent module at the command line before telling SurgeMail to use it!!

Configuring the Authent Module.

Normally you configure the authent module through the admin interface, but if you find yourself editing the surgemail.ini by hand ensure you pass the -path command line parameter to the authent module, this is to tell it where to find its config file and any other files it might use, for example:

```
g_authent_process "c:\surgemail\nwauth.exe -path c:\surgemail"
```

The above tells NWAAuth to look in c:\surgeemail for it's files nwauth.add, nwauth.txt, etc.
The same is true for any module that has an .ini file.

If you're authent module is not working this is the most likely cause.

Extended info fields recognized by SurgeMail

SurgeMail uses the [g_authent_info](#) settings to define what fields it displays and where. Most fields have a 'hard-coded' use but others are simply there as examples of the kind of optional information you can collect about your users. The default settings are as follows:

```
g_authent_info name="Creation Stamp" field="created" access="none" default="" type=""
g_authent_info name="Forwarding" field="fwd" access="none" default="" type=""
g_authent_info name="SPF Block" field="spf_block" access="none" default="" type=""
g_authent_info name="Disk Quota (bytes)" field="quota" access="domadmin" default="" type=""
g_authent_info name="Full Name" field="full_name" access="user" default="" type=""
g_authent_info name="Phone" field="phone" access="user" default="" type=""
g_authent_info name="Password Retrieval Question" field="pass_question" access="createonly" default="" type=""
g_authent_info name="Password Retrieval Answer" field="pass_answer" access="createonly" default="" type=""
g_authent_info name="Access type" field="mailaccess" access="domadmin" default="" type=""
g_authent_info name="Account Status" field="mailstatus" access="domadmin" default="" type=""
g_authent_info name="Sms Number" field="smsto" access="domadmin" default="" type=""
g_authent_info name="Disabled" field="disabled" access="none" default="" type=""
g_authent_info name="User alias quota" field="alias_quota" access="domadmin" default="" type=""
g_authent_info name="User list quota" field="list_quota" access="domadmin" default="" type=""
g_authent_info name="User access settings" field="user_access" access="domadmin" default="" type=""
g_authent_info name="Msg limit per 30min" field="send_limit" access="none" default="" type=""
g_authent_info name="To host(g_proxy)" field="tohost" access="none" default="" type=""
g_authent_info name="Is an alias of" field="realuser" access="none" default="" type=""
g_authent_info name="Allowed to" field="allow" access="none" default="" type=""
g_authent_info name="Friends Enabled" field="friends" access="none" default="" type=""
g_authent_info name="Email Notification Address" field="enotify" access="none" default="" type=""
g_authent_info name="SpamPrivate private prefix" field="ddpriv" access="none" default="" type=""
g_authent_info name="SpamPrivate from prefix" field="ddfrom" access="none" default="" type=""
g_authent_info name="Card Name" field="ccname" access="user" default="" type=""
g_authent_info name="Card Number" field="ccnumber" access="user" default="" type="encrypt"
g_authent_info name="Card Expiry" field="ccexpires" access="user" default="" type=""
g_authent_info name="Card Security Code" field="ccciv" access="user" default="" type=""
g_authent_info name="Card Type" field="cctype" access="user" default="" type=""
```

Each field is used for a different purpose:

allow	Services the user can access eg. SMTP,POP,IMAP.
created	Record of creation time, stored on creation time.
ddfrom	Private email 'from' suffix.
ddpriv	Private email 'private' suffix.
enotify	The email address to send email notifications to.
friends	'true' if the user has a friends mode configured.
full_name	Example information about user (not required, example).
fwd	Forwarding rules for the user, configured via users "Forwarding" page.
mailstatus	Status of the account, see (account status)
pass_question	Only used at creation time, collects password retrieval question (not stored in database).
pass_answer	Only used at creation time, collects password retrieval answer (not stored in database).
phone	Example information about user (not required, example).
quota	Users disk quota, configured via the administrative interface.
spf_block	'true' if the user wants to block non spf compliant email.

For example:

```
+OK bob@test.com config 0 fwd="fred@test.com"
+OK bob@test.com config 0 quota="200000" fwd="joe@xx.com"
```

Advanced settings :

alias_quota	Number of aliases this user can create
admin_access	Features this domain admin can access
ccname	Credit card holders name.
ccnumber	Credit card number.
ccexpires	Credit card expiry date mm/yy.
ccciv	Credit card security code.
cctype	Credit card type eg. Visa, Amex
disabled	Used by email based account creation code (may also be used to disable existing accounts)
list_quota	Quota of mailing lists the user can create.
mailaccess	Used in conjunction with g_access_group and g_user_access to specify access to features.
realuser	Real account to which this account is aliased - allows aliases to be specified in authent database
send_limit	Number of outgoing messages this user can send per 30 minutes. You must also define the global limits g_tarpit_max, and g_tarpit_max_remote. And you may want to set g_tarpit_drop "true"
smsto	SMS phone number to send SMS notifications to users "SMS" page.
tohost	The host which to connect to when using proxy mode (g_proxy)
user_access	Features this user can access

Legacy settings :

accountstatus	Numeric equivalent of mailaccess
droppath	The user's drop path, this is no longer supported and will not work with all SurgeMail functionality.
groups	Example setting used to be installed for default SurgeMail installs

Mixed case usernames and domains

SurgeMail will lowercase domains in all cases, and for usernames and passwords entered in mixed case it will attempt a lookup 'as is' and then a second one using lowercase, this helps avoid problems with users accidentally mixing case.

In all cases drop paths etc are created using lowercase as this avoids the terrible mess on UNIX that can occur. This does mean it is impossible to have two different users who are only distinguished by case. This is of course an intentional feature and not a bug. We think anyone who actually wants multiple users with the same name is a little crazy :-)

Example configuration of firstname and lastname for LDAPAuth

This is an example of how to configure LDAPAuth for SurgeMail such that the user **must** enter a first and last name upon creation. This is how you might configure it for use with the Thunderbird LDAP client.

First, in ldapauth.ini add/change:

```
info_fields full_name cn
info_fields firstname givenName
info_fields lastname sn
```

and remove:

```
must_set_fields sn name
must_set_fields cn name
```

Next, in surgemail.ini configure:

```
g_authent_info name="First Name" field="firstname" access="user"
g_authent_info name="Last Name" field="lastname" access="user"
```


and for each domain configure:

```
create_reqd "firstname,lastname"
```

Then stop and restart the surgemail process. Users will now see two fields additional on the self creation page "First Name" and "Last Name" the data entered here will be stored in the LDAP fields specified "givenName" and "sn" and Thunderbird will use these values.

Example config used with ActiveDirectory (Windows)

```
ldap_host 10.1.1.1
ldap_port 389
ldap_mgr_dn cn=ftpadmin1,ou=mgt_info_sys,ou=CTL,ou=region_sales,dc=example,dc=com
ldap_mgr_pw secret_password
ldap_search_base OU=region_sales,dc=example,dc=com
ldap_scope LDAP_SCOPE_subtree
ldap_search_name ExampleAccountName
ldap_group_base OU=region_sales,dc=example,dc=com
ldap_group_search CN=&*
ldap_group_field CN
ldap_group_attr member
```

Virtual Domains - a guide

Unlike web servers there are two basic types of virtual domains for a mail server:

- Real IP based virtual domains, where you have allocated an IP address to each virtual domain, the server can use this information to figure out which domain it should 'pretend' to be.
- Fake ones, where you only use a single IP address, then the user must login as 'user@domain.name' when fetching their Email via POP so that the server can figure out which domain they belong to.

SurgeMail supports both of these methods or even combinations of them and any number of virtual domains. However, it supports some other systems too :-). For example, you can tell SurgeMail to respond to all domains matching a specific wild card eg: *.mydomain.com You can also use a virtual user table where each user is in a 'domain of their own' (this is useful if you want to sell users their own domain names).

In addition, some mail clients do not allow a user to specify user@domain.name as their 'username'. In this case you can define a domain separator like '/' and then the user can login as 'user/domain.name'

How to create a virtual domain

Simply click on 'domains' 'add' in the web admin tool. Then fill out the details.

In addition if you are using WebMail you may need to add these domains to webmail.ini as [described](#).

Note: SurgeMail will not create the directory structure until a message is received for the new domain, so don't panic if you can't see the domains directory right away.

You can define default settings for new domain by creating/editing the domain_defaults.txt file in the SurgeMail "web" directory (if you create/change this file manually copy it to the mirror too). This file contains only basic settings in the form:

```
<setting_name><space><setting_value>
```

one setting per line. It cannot handle the more complex settings eg.

```
setting label="value" label2="value2"
```

Adding IP numbers to your operating system

It is fairly easy to add multiple IP numbers for a single machine, up to 255 per interface is fairly straightforward. 1024 is usually possible with minor patches. The exact method varies. On NT just specify extra numbers in the networking control panel.

For UNIX, see <http://www.nethelp.no/net/vif/readme.html> for more information.

As an example, on Linux you would do the following:

```
su - root
ifconfig eth0:2 999.59.4.31 up
to add a second IP number 999.59.4.31. The number :2 can be anything between :1 and :255
```

Virtual Domain User (domuser.dat)

If every user on your system is in their own domain, e.g. bob@jones.mail.com, john@smith.mail.com then you probably don't want to create a virtual domain entry in surgmail.ini for all 200,000 users. Instead you can use the file domuser.dat to list each user so that SurgeMail knows what virtual domains exist on your system.

The format is:

```
delivery_user_domain user_database_lookup_name
```

e.g.

```
xxx@bob.com xxx@bob.com
yyy@yyy.com yyy@bob.com
*@ccc.com ccc@bob.com
```

So with the above file, if a user 'fred@ccc.com' logged in they would be looked up in the userdatabase as 'ccc@bob.com' and yyy@yyy.com would be looked up as 'yyy@bob.com'

This file is very efficient and can support millions of virtual domains on a single server.

MX records DNS entries etc...

DNS stands for "Distributed Name Server", and it is the mechanism whereby your.domain.name is translated into your IP number (e.g. 13.2.44.2 etc). When you setup a mail server you will need to add a DNS entry. Normally this is done when you register a domain. Your domain registration service should also let you specify the IP number that your computer uses.

In addition, they may let you specify MX records. These are used as an over-ride, as a mail server for xxx.com may not run on the same computer that handles web pages for xxx.com. Also for redundancy you can specify 2 or more computers to receive Email for your domain, so if one system is down the other will accept the mail and hold it until your main system is up.

All the following examples really only apply to you if you run your own DNS server. Most likely someone else is running one for you (your ISP or domain registrar), so they will be adding these entries, but you still need to understand these entries so you know what to ask them to add to their DNS server.

For your mail server to work you must at least have a DNS or MX entry for your system e.g. typical entries look like this:

```
mail.freemail.com. in a 10.0.0.12
```

Alternatively, or as well, you can use MX records. Typically these are used to give a backup mail server address, e.g.

```
mail.freemail.com. mx 10 mail1.freemail.com.
mail.freemail.com. mx 20 mail2.freemail.com.
mail1.freemail.com. in a 10.0.0.12
mail2.freemail.com. in a 10.0.0.13
mail.freemail.com. cname mail1.freemail.com.
```

This says send all Email to mail1.freemail.com if possible. Failing that, send to mail2.freemail.com. Then the next two lines give the IP addresses of those two systems, and the last line is for dumb systems that don't know how to do MX lookups so they will find the primary system anyway.

Often you would setup the low priority MX entry to point to a system outside your local LAN, but if so, you must ensure they are configured to allow forwarding of Email to your system. Then while your system is down they will collect incoming Email for you.

Lastly you may want to configure wild card entries if you have lots of virtual domains all under a primary domain name, e.g.

```
*.freemail.com. mx 10 mail1.freemail.com
*.freemail.com. cname mail1.freemail.com
```

Beware - wild card entries are not as simple as they look. In particular, it cannot be guaranteed if the MX or IN entry will be used.

You should also specify a reverse DNS entry for your mail server. If you don't some other mail servers might treat you as a spammer and block or ignore all your Emails.

When mail.your.domain and your.domain are NOT the same system

The basic problem is you want user@your.domain to go to your mail server, but you want http://your.domain to go to your web server, and they are different systems.

This is achieved by the following magic incantations.

- Add MX records to point your.domain to mail.your.domain
- Add aliases for the domain in surgemail.ini so it will accept mail for mail.your.domain as well as your.domain, host_alias "mail.your.domain" (in each domain)
- Add g_server_name url="*.domain.com" name="domain.com" so that web connections will figure out the right domain
- In webmail.ini the imap server etc should be defined as 'mail.your.domain' not 'your.domain'

So in your DNS you have:

- mail.your.domain --> x.x.x.x (your mail server)
- your.domain --> y.y.y.y (your web server)
- your.domain MX --> mail.your.domain

When you add a domain in SurgeMail it will ask you about your 'DNS' and 'MX' names, if you specify them as different, it will correctly setup your webmail.ini and surgemail.ini to cope, but you still need to setup

the correct DNS/MX records in your DNS server.

Domain identification using WebMail with SurgeMail

Webmail combined with surgemail can host multiple domains on one server. There are several options for having webmail identify the correct domain.

1. Web browser URL based (most commonly used)
2. Domain dropdown list
3. Fully specified login
4. Using IP address

1. Using the web browser URL

The normal way of handling multiple domains is to login to webmail using your web browser on a URL of `http://mydomain.com/scripts/webmail.exe`. Webmail will then display a login page that has the domain correctly identified. For this to work this domain needs to be configured in several places in surgemail and webmail.

The vdomain in question must be defined in surgemail:

```
surgemail.ini:
vdomain address="" name="mydomain.com"
mailbox_path "C:\surgemail\mbx\mydomain.com"
{... other settings}
```

and in webmail:

```
surgemail/web_work/surgehost.ini:
vhost mydomain.com
domain mydomain.com
suffix @mydomain.com
imaphost 127.0.0.1
smtp host 127.0.0.1
vend
```

The definition of the domain in both places is generally done automatically by surgemail when you create this domain using the surgemail web admin interface or tellmail commands, but can be manually added if required.

When the browser connects to surgemail on the url **`http://mydomain.com/scripts/webmail.exe`**, it passes "Host: mydomain.com". SurgeMail matches this host value against `g_server_name` settings, domain names, `url_host` values and also compares the ip address of the domain to the ip being connected to. When it finds a match it passes a **SERVER_NAME** environment variable to WebMail with the appropriate value. WebMail matches this value against the vhost sections, identifying the domain.

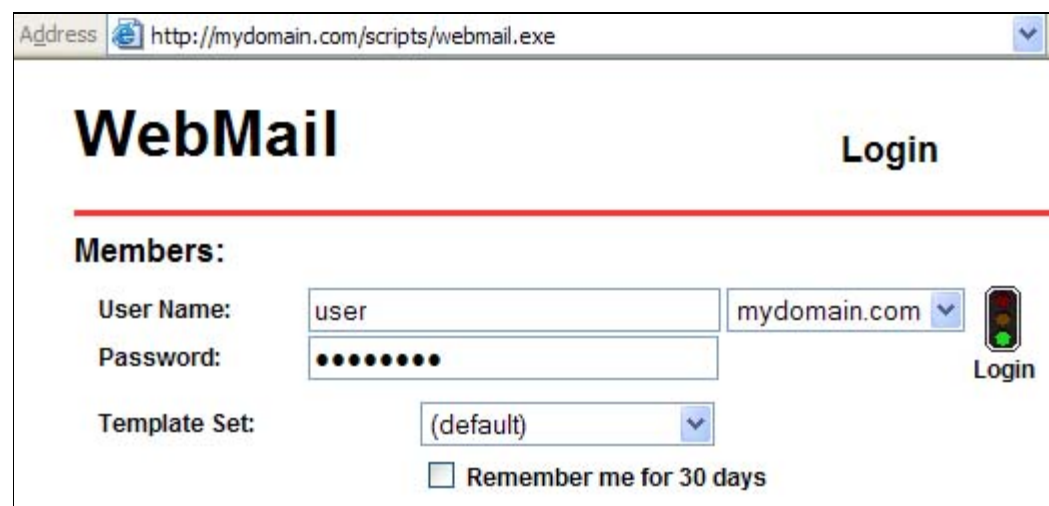
If your url does not match your vdomain name eg you want your users to connect to

`http://webmail.mydomain.com/scripts/webmail.exe` to send email of `user@mydomain.com`, surgemail will need to do this name translation. If the domain has a unique ip, and the url domain resolves to that ip then it should automatically work (surgemail 3.7a-21 onward). If you have only 1 such url domain name then you can set the domain `url_host` setting to this value (this has the side

effect that SurgeMail will also use this value in any url it generates internally). If you have several values you can use a g_server_name rule, this rule may contain a wildcard. eg.

surgemail.ini:
g_server_name url="*.mydomain.com" name="mydomain.com"

2. Domain dropdown list



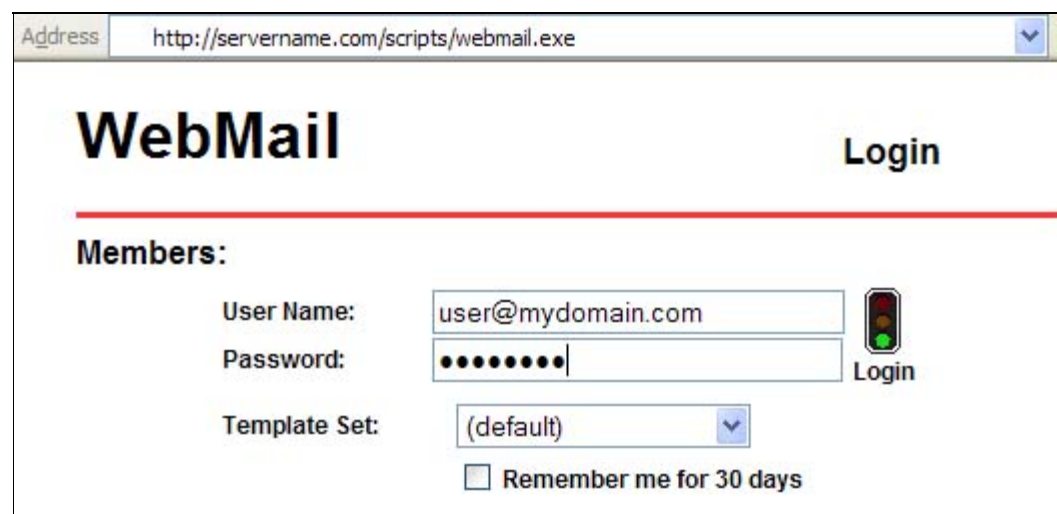
The screenshot shows a web browser window with the address bar displaying `http://mydomain.com/scripts/webmail.exe`. The page has a header with "WebMail" on the left and "Login" on the right. Below the header, there is a section titled "Members:". Under "Members:", there are three input fields: "User Name:" with the value "user", "Password:" with masked characters "••••••••", and "Template Set:" with a dropdown menu showing "(default)". To the right of the "User Name:" field is a dropdown menu showing "mydomain.com". Below the "Template Set:" field is a checkbox labeled "Remember me for 30 days". To the right of the "Password:" field is a "Login" button with a traffic light icon.

An alternative to having webmail identify the domain using the web browser URL is to always display a dropdown list of all domains. This has the disadvantage that all users will see all domains that your server hosts. To display a dropdown in the webmail and user.cgi pages use the following settings:

surgemail.ini:
g_user_domainlist "user"

webmail.ini:
domain_select true

3. Fully specified login



The screenshot shows a web browser window with the address bar displaying `http://servername.com/scripts/webmail.exe`. The page has a header with "WebMail" on the left and "Login" on the right. Below the header, there is a section titled "Members:". Under "Members:", there are three input fields: "User Name:" with the value "user@mydomain.com", "Password:" with masked characters "••••••••", and "Template Set:" with a dropdown menu showing "(default)". To the right of the "User Name:" field is a "Login" button with a traffic light icon.

An alternative to 1) and 2) above is to display an edit box allowing a fully specified login. To setup webmail this way remove the text "@|domain|" from the login.tpl file for the template set that you are using.

Automatic Webmail login

If you wish to automatically login to webmail from another web page / application you have three options:.

1. Pass username and password (not generally a good idea)
2. Pop based autologin (surgemail only)
3. File based autologin

1. Pass username and password

You can just pass the username and password as fields to webmail login page. This means the password gets sent to the web browser before submission so the password is sent unencrypted across the (possibly unsecure) network. This could even get stored in web server referrer logs if you are unlucky.

2. Pop based autologin

Use POP based autologin (surgemail mailserver only) This is the default that surgemail uses when logging in to webmail and entails surgemail remembering the encoded password and serving this when webmail asks for it as an extended POP command.

This requires:

```
surgemail.ini:
g_autologin_pop "true"
webmail.ini:
use_id_autologin true
```

3. File based autologin

Using file based autologin allows arbitrary applications to login directly to webmail without sending the password in plaintext to the web browser.

The application that is doing the auto login to surgemail stores the password as an encoded file and sends a token to this file in the url it sends the web browser. Webmail reads this file, stores the password and removes this temporary autologin file. This means the password is not sent across the (possibly unsecure) network.

This requires:

```
webmail.ini:
use_id_autologin false
```

and if surgemail autologin is also required:

```
surgemail.ini:
g_autologin_pop "false"
```

1 - The "central login" application saves the encoded password in a file (encoded as per following logic) in the webmail work folder:

In file 22413.tmp (where 22413 is arbitrary random integer) store the password passed through the auto_login_encode function.

```
char *auto_login_encode(char *name,char *pass,char *enc,int bfsz) {
    // bfsz = size of enc buffer. (Typically 1000 bytes)
    char *n;
    int i;

    for (i = 0,n = name; i < bfsz-1 && pass[i]; i++) {
        enc[i] = pass[i] + *n++;
        if (!*n) n = name;
    }
    enc[i] = '\0';
    return enc;
}
```

2 - The following url is then passed to the browser, where the critical fields are user, id and auto_login:
http://myserver.com/scripts/webmail.exe?

cmd=auto_login&user=marijn&select_domain=mydomain.com&id=22413&frames=true

3 - Webmail uses the password from the 22413.tmp autologin temp file and removes the 22413.tmp file.

Surgemail Clustering Architectures

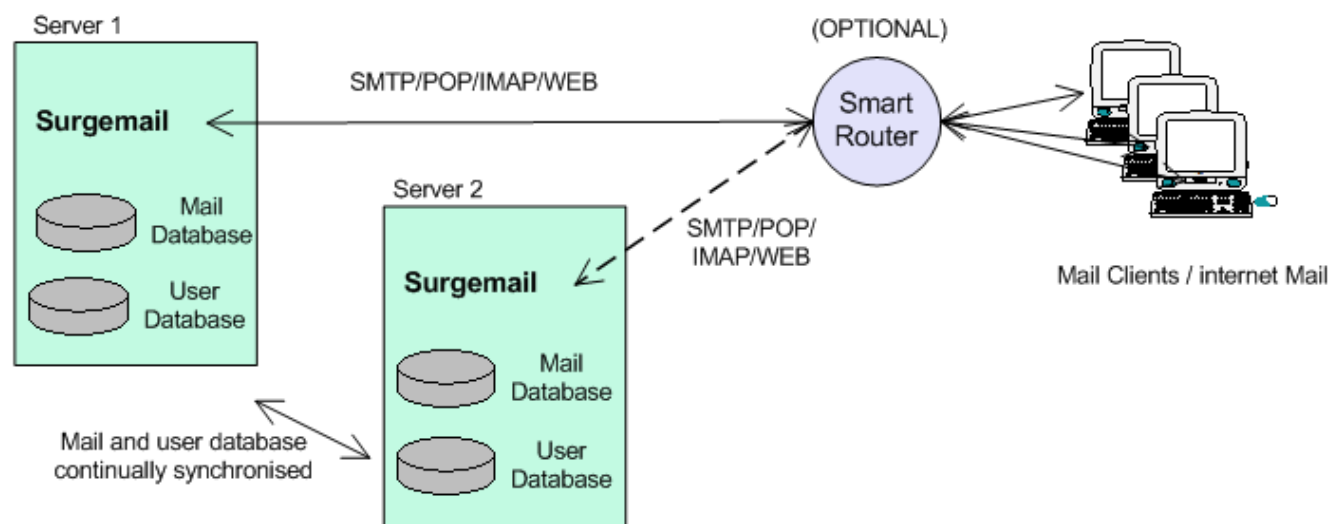
To achieve very high reliability, redundancy or scalability, surgemail can be configured using a combination of several clustering architectures. Each clustering architecture has its own advantages and tradeoffs which you must consider in relation to your business need. The clustering architectures include:

- [live replicate server using mirroring](#) (recommended)
- [functionally split across several servers](#) (recommended)
- [traditional shared storage cluster](#) (not recommended)
- [domain split across several servers using proxy mode](#) (not recommended)
- [three layers model using proxy mode](#) (not recommended)
- [choosing which one to use](#)

Live replicate (Mirroring)

Using surgemail mirroring you can setup two servers to be continually updated "live replicates of each other" allowing you to send mail in to either system and read mail back from either server. In this configuration there is no single point of failure and if there is a major hardware problem on either server, you can failover to the second server with no interruption of service. Also, if one system goes down for maintenance, it will auto-resynch when it comes back online.

Mirroring is the simplest and most cost effective way of getting a system with high reliability and high redundancy. This is particularly useful if your mail load "can easily" be handled on a single server.



This failover can be done a variety of ways, but the recommended ways is either to use router based failover or manually switched using an extra floating IP address that is allocated to the primary server.

- **Router based failover** If you are using router based failover it is recommended that you configure the router such that all consecutive connections from a single IP address get connected to the same server. This does not matter for POP and SMTP but is important for IMAP (and Webmail) connections. Alternatively configure the router so that all connections go to one server and failover to the second server if the first server stops responding for a period.
- **Manual IP switching** Some people prefer to have more control over the failover, or do not have routers capable of hardware failover. In this case point your DNS record to an extra floating IP address that you manually allocate to one or other of the mirrored servers.
- **Other** Note there are some alternative options for failover the use of which is discouraged, such as switching DNS records (long delays) or using [scripts](#) to switch floating IPs (not reliable).

Considerations:

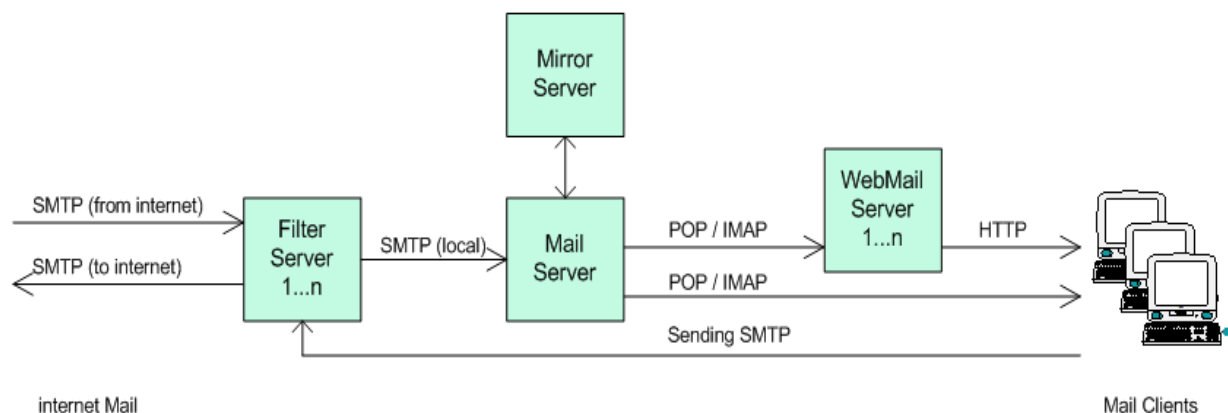
For more information on configuring mirroring see [Mirroring FAQ and Configuration examples](#) help page.

Functionally split

Surgemail can be functionally split across several servers. The main reason to use this is if your mail load is too large for one server (eg 40000 user+) and / or you have a particularly heavy spam loading or webmail client loading.

You can pick and match what you want to support on each server, but typically you would setup say 2 front end systems for spam and virus filtering. A single mail system to handle storage of local mail including access to this using POP and IMAP. And one or more webmail systems which handle the webmail load and talk to the primary mail server when necessary using IMAP.

This is the most efficient way to implement a high reliability system with a high level of scalability. Dependant on your user needs this allows you to host up to 100,000 users on your primary mail server..



Other considerations:

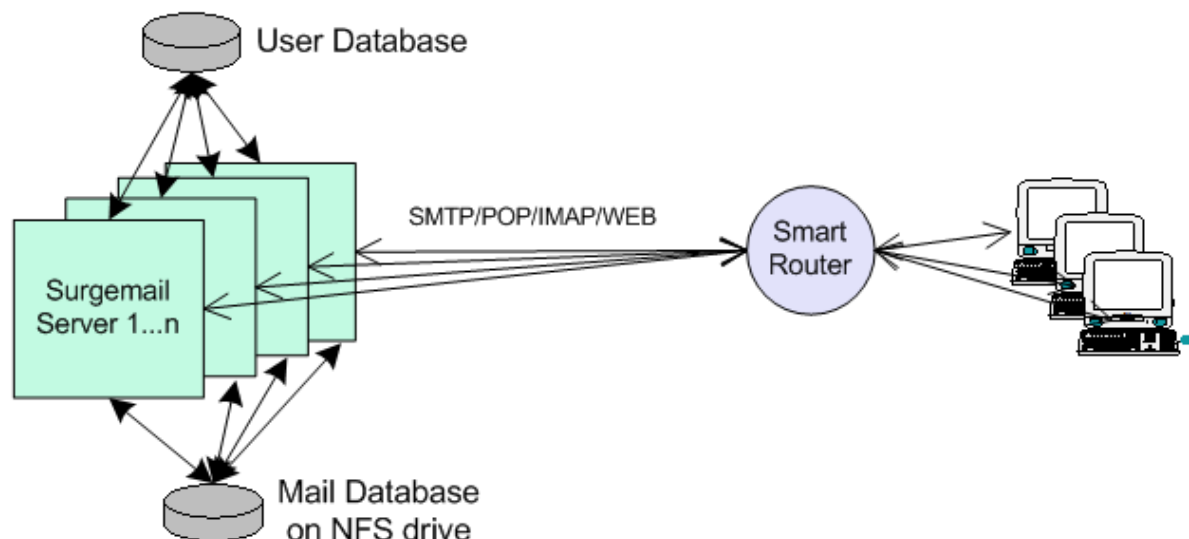
- A functionally split architecture can be combined with mirroring. You would simply introduce one more system into the above architecture which is a mirror of the primary mail system. As per mirroring this removes the single point of failure (with associated mail data loss) you would otherwise have if your main mail system were to fail.
- Mail will continue to be accepted by the filter systems if there is a problem with your primary mail system.

For detailed configuration information see [configuring functionally split cluster](#) help page

Shared storage cluster

Surgemail can be configured in a more traditional shared storage cluster configuration using an NFS (or other) shared storage device for providing standard mail services.

In this configuration you have several servers all running surgemail handling all mail services storing users mail using the same central storage. The incoming connection load is shared between all servers using an appropriate technique. This is typically a hardware based load balancing router.



This configuration has the advantage that it is truly symmetric and you can easily add in one or more servers if required. However the shared storage cluster configuration has two significant disadvantages:

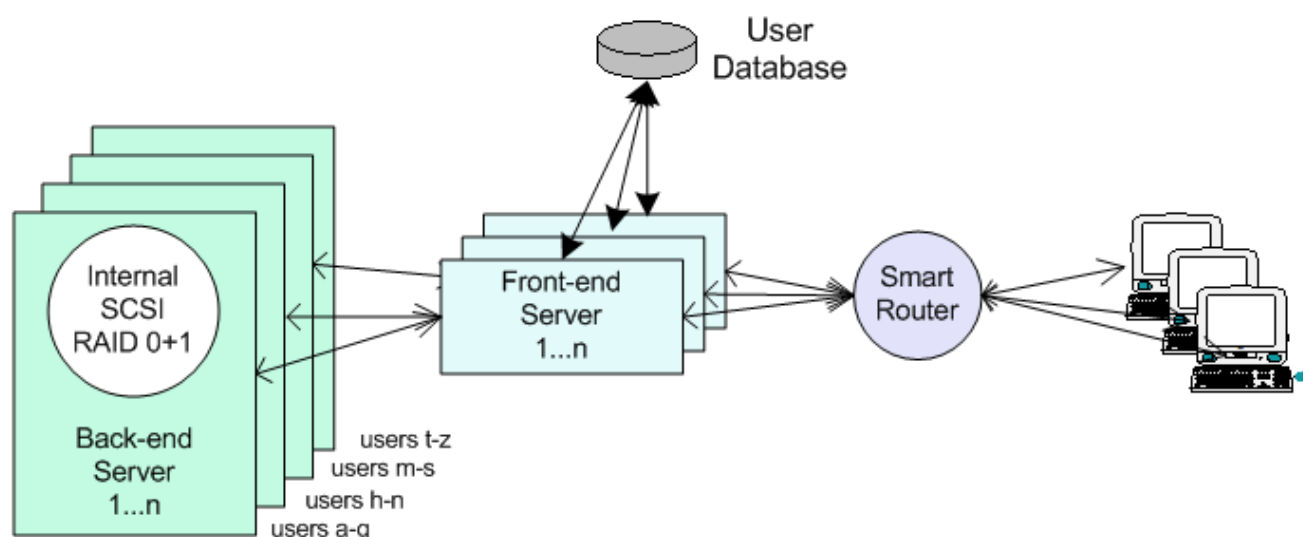
- 1) Less efficient - Several, normally in memory optimisations (in particular quota handling and file locking) needs to be done on disk, increasing the disk IO load.
- 2) Some of surgemail advanced features are not fully functional (eg surgeplus calendaring)

For detailed configuration information see [configuring shared storage cluster](#) help page.

Domain split (Proxy mode for huge systems)

Proxy mode allows a domain to be split across several physical servers. This systems allows both infinite scaling, and 3 layer security. Incoming POP/SMTP connections arrive at one of several front end 'proxy' servers (running SurgeMail in proxy mode) these servers then lookup the user in the networked user database (via LDAP or our own TCPAuth module) and along with the normal response an extra response code of 'tohost=backend.host.name' is returned, the proxy then redirects the user to the appropriate back end system.

So you might run 4 back end systems, each with 100,000 users, and 2 front end systems. To add more users you just add as many front end and back end servers as needed to cope with the load.

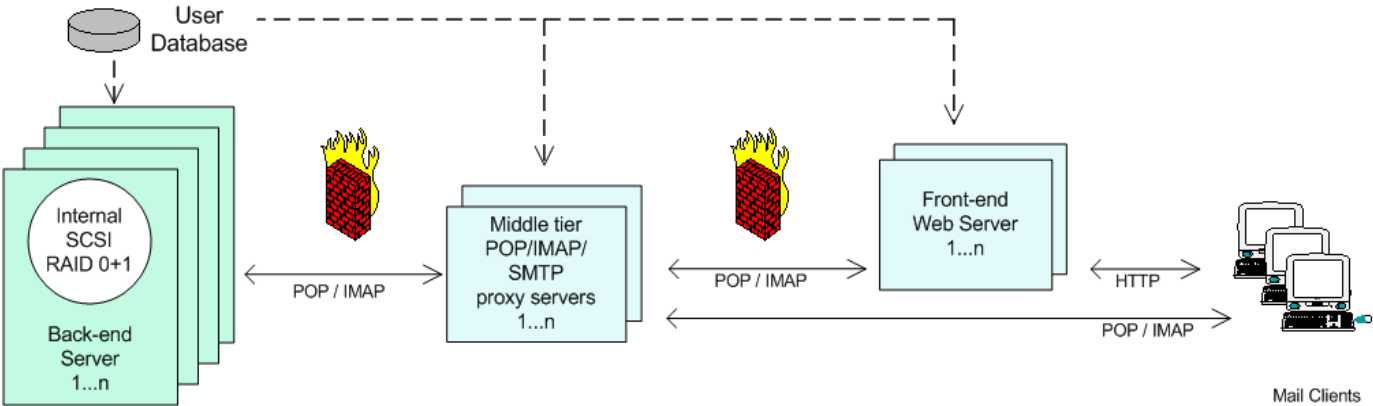


Each user is only on one of the back end systems, the only piece in the system that has to handle all the users is the user database, which is a relatively trivial task as the quantity of data per entry is so small. We recommend the use of NWAAuth or LDAPAuth but any of the database back end authent modules would be suitable.

For detailed configuration information see [configuring proxy mode cluster](#) help page.

Three tier model (Proxy mode for increased security)

In the tree layer model, proxy mode is used to split the system in to three tiers (separated by a firewall), each with a different security level. The top layer of servers exposes webmail directly via http. The middle tier exposes POP / IMAP via proxy systems and the backend systems are not directly exposed to the network at all.



In

particular some telcos require this structure to their mail system.

Combination & tradeoffs

As already noted a combination of the above can also be used. Typical examples that you might use:

- 1. Functionally split cluster + backup of mail system using mirroring
- 2. Proxy mode split cluster + backup of each backend mail system using mirroring

By themselves the clustering techniques compare as follows:

	Mirror	Functionally split	Shared Storage	Domain split (proxy)
Provides processing redundancy	Some	Yes	Yes	No (but can be added by splitting to functionally split or shared storage clusters)
Provides data redundancy	Yes	No (but can be added using mirroring)	No (but can be added using mirroring)	No (but can be added using mirroring)
Provides load sharing	Some	Yes	Yes	Yes
Provides for incremental upgrades	Some	Some	Yes	Some
Use of basic mail features				
SMTP	Yes	Yes	Yes	Yes

POP	Yes	Yes	Yes	Yes
IMAP	Yes	Yes	Yes	Yes
Webmail	Yes	Yes	Yes	Yes
Use of advanced features:				
Surgeplus Filesharing	Yes	Yes	Yes	Yes
Surgeplus Calendaring	Yes	Yes	Some*	Yes*
Mailing Lists	Yes	Yes	Yes*	Yes*

* Some special conditions apply. eg functionality may need to be setup on only one particular system in the cluster or not all of the advanced functionality may be available.

Mirroring (Server Replication) FAQ

- [What is mirroring](#)
- [Should I be setting up mirroring in my environment or is it a bad idea?](#)
- [How do I turn it on ?](#)
- [Do not load balance mirrored servers](#)
- [Can I mirror the config file as well ?](#)
- [Adding a mirror to an existing system](#)
- [What's the deal with master/slave, can I swap them over?](#)
- [How do I know it is working? Is it in sync yet ? Is it keeping up ?](#)
- [How long can I expect mirroring actions to take in a real environment \(100 user + 1GB mail / 5000 user + 50GB mail / 100000 user + 500GB mail\)](#)
- [What happens if I stop surgmail / surgmail crashes / change my mirroring configuration etc. halfway through some of these mirroring actions.](#)
- [If mirroring is really such a good idea why do none of the competing products offer this capability.](#)
- [I have two live mail system that I would like to mirror each other. Can I and / how do I configure this?](#)
- [Can I check it's always working](#)
- [Is there any functionality / settings etc. that are not mirrored?](#)
- [Performance](#)
- [How to swap master/slave when one system dies or is replaced...](#)
- [Move SurgeMail to a new system \(and or change operating systems\)](#)

Mirroring the server - What is mirroring.

The SurgeMail 'Mirror' system allows you to link two systems together and read or deliver Email to either system and both systems will continually 'match' each other. This can be used in several ways:

- Keep a live backup system for 'hot swapping' in case of failure or upgrade requirements on your live system.
- Move a system from one geographic location to another with no downtime

Mirroring will work over a LAN or WAN connection and can be encrypted. Unlike using shared NFS drives there is no single point of failure in a SurgeMail Mirrored system so you have genuine fail over capability.

Should I be setting up mirroring in my environment or is it a bad idea?

In almost all cases, you should be running a mirror of your mail server, it's the cheapest and most efficient way to keep a live backup of your system. The only cases we can think of where you don't need a mirror are if:

- You are running a home system, and you don't mind if you loose your mail folders.
- You have a working daily backup of your mail store, and you are happy to lose up to 24 hours of email when your disk fails.
- You are running a free service and you have warned your customers that the service may vanish 'at any time'

Some people forget that disk drives fail, they do, your mail server's disk will fail approximately once in the next 2-3 years. Some people think RAID 5 or similar systems provides protection from disk failure, it does not, we've had so many customers loose Raid 5 arrays (and we've lost so many) that we actually consider them less reliable than non raid5 disk arrays. (Speaking of which always use RAID 10 for high performance and reliability for a mail server, when possible, NOT Raid 5)

Do not LOAD BALANCE mirrored servers

Many people think mirroring should be combined with a load balancer as you would do with a web server, this is NOT the case. A simple load balancer causes serious risks when used with mirroring because if the mirroring fails even briefly, and the user accesses both of the servers during that time, the new messages could be assigned identical UID values. Then one of those messages will be invisible and lost to the end user.

To avoid this and still have a fault redundant system you can do any of the following

- Only deliver new mail to one host.

- or Only failover rather than load balance
- or Make the load balancer balance on incoming ip address when it can so it doesn't randomly move users between hosts.

Authent modules that support mirroring.

In general nauth is the only module that natively supports mirroring, but some other modules work where they can both access a common back end server (like mysql, ldap etc) nauth doesn't work because it relies on some local files to fill in fields that are not available in the windows database.

Module	Support?
nauth	Yes
ntauth	No
mysqlauth	Yes but both servers must point to the same mysql database backend
ldapauth	Yes again both servers must point to the same ldap back end typically

How do I turn it on?

In brief:

1. Copy surgemail.ini to the slave system and adjust any system specific settings.
2. Turn on the mirror settings on both servers, set mode to "master" on the master, and 'slave" on the second system.
3. If you are mirroring configuration then issue tellmail resync_config on master system
4. Then issue tellmail resync_fast on master system.

If you want to add mirroring to an existing server, you'll need to read [this](#).

Simply setup two mail servers in a similar manner, we recommend you copy the config from one to the other and then adjust any system specific settings (mail paths etc.) it's important that the configs have the same domains and forward rules and the same g_mirror_secret)

Example: (adding these settings to surgemail.ini)

Server 1: ip 10.0.0.1 (master)

```
g_mirror_nossl "TRUE"
g_mirror_mode "master"
g_mirror_host "10.0.0.2"
g_mirror_secret "testing"
g_mirror_config "true" (if you want to mirror config changes as well)
```

server 2: ip 10.0.0.2 (slave)

```
g_mirror_nossl "TRUE"
g_mirror_mode "slave"
g_mirror_host "10.0.0.1"
g_mirror_secret "testing"
g_mirror_config "true" (if you want to mirror config changes as well)
```

So above are the settings that go into each servers surgemail.ini. That will give you a mirror, its that simple.

Now you need to consider how users get to the server and how you can easily allow them to get to the 'working' server in the event of a failure.

For incoming messages you can just setup 'MX' records so that the backup server is listed as a low priority host. e.g.:

```
your.domain MX=10 mail.your.domain
your.domain MX=20 mail2.your.domain
```

But for user access to the server you have several options:

1. Tell users to use 'mail2.your.domain' in the event of a failure (suitable for office mail servers)
2. Manually change the IP number of the systems in the event of a catastrophic failure.
3. Invent an 'extra' IP number for the mail server and assign it to the 'working' box. Then manually add that IIP number to the other system during a failure of the main system.

4. Use system 3, but then use some scripts to automatically change the IP number during failures (not recommended) [See here for details](#)
5. Use system 3 but then use a router that can do the failover on the fly for you. (expensive but reliable)

Using config setting mirroring (Requires SurgeMail 3.1 or later)

You can choose to enable config setting mirroring. This causes SurgeMail to send it's config from master to slave and vice-versa if/when config changes are made in the web interface (it does not notice manual changes done by editing the config file).

First make a backup of both ini files, just in case :-)

To enable it set:

```
g_mirror_config "TRUE"
```

on both machines, then execute:

```
tellmail resync_config
```

on the machine which has the desired config.

Of course, you do not always want to mirror all the settings, especially settings to handle mirroring like `g_mirror_host` for example. You may use [g_mirror_config_except](#) specify settings to *be ignored when processing an incoming config*, in addition there are a number of settings which are ignored by default, see [g_mirror_config_except](#) for details.

Adding a mirror to an existing system

You need to install surgemail on a new system, then follow the instructions above in "[How do I turn it on?](#)" to add the correct mirror settings to both ini files (old system and new system), you should set the new system up as SLAVE, then...

```
issue "tellmail resync_config" on master (if using g_mirror_config)
issue "tellmail surgehost_update" on master
issue "tellmail resync_nwauth" on master (if using nwauth)
issue "tellmail resync_fast" on master
```

Note: Although this feature exists in earlier versions of surgemail, we recommend upgrading to 3.1 before using it as we made significant improvements to the fault tolerance of this feature (it's more idiot proof in version 3.1 :-)

How do I know it is working? Is it in sync yet ? Is it keeping up ?

Always check in two ways, first check the status as below, then compare two directories manually to be 'absolutely' sure.

In the status window (near the end) you will see the following information

Mirror out: Que/sent add=612/611 (3343432 bytes) del=612/610

Mirror in: Received add=0 del=0 rename=0 failed=0

This shows both halves of the mirroring operation. The "Mirror out:" line shows messages queued to be sent to the other system and the second number (/611) shows how many have been successfully sent (so one is still queued) and how many delete operations have been queued (612) and how many have been sent (/610), so 2 are still queued. Obviously these numbers should normally match.

The second line, "Mirror in:" shows how many new items or deletions have arrived from the other system.

To compare directories do this on both servers, and compare the directory listings:

```
tellmail path user@domain.name
dir [path it returns]/mdir/new
```

Lastly, issue a 'tellmail resync_fast' and check in the status to see how many corrections it needs to send.

What's the deal with master/slave, can I swap them over?

For internal reasons we needed to establish a master/slave concept, although in almost all respects they are identical and neither is the 'master' in any behavioral sense, for example if you change something on the slave the

change will appear on the master and versa visa. The one thing you should never do is swap the master/slave settings over as this will confuse the mirroring software! (It can be done reasonably safely if both servers are stopped at the time, but it's best avoided :-)

We do recommend that you generally avoid doing things on both servers randomly, it's best to make everything go through one server and do all changes etc. on one server, then use the other server purely as a 'hot' backup. In this way if something does go wrong, but goes unnoticed (e.g. they get unplugged from each other) you will know which one is in a 'good' state and which one is 'out of date'

Note: DLIST runs only on the 'master', so in the situation where your master is going to be down for several days, you will need to swap master and slave so that the dlist on the 'slave' will come to life.

How long can I expect mirroring actions to take in a real environment (100 user + 1GB mail / 5000 user + 50GB mail)

How long is a piece of string :-), the time is mainly dependent on the 'number' of messages stored on the server, so it is not directly related to the number of users, or the size of your mail store.

But as a rough guide, to resync from scratch, you would expect it to take something like:

- 10 minutes for 100 users first time, 1 minute to resync
- 3 hours for 1000 users first time, 4 minutes to resync
- 3 days for 40,000 users first time, 3-4 hours to resync.

What happens if I stop surgemail / surgemail crashes / change my mirroring configuration etc. halfway through...

The mirroring is very forgiving, it will try to continue after a crash, when one server is down changes are 'queued' until it reappears. The only time you must issue commands is when one server's disk is lost/reformatted, then you must issue a 'tellmail resync' on the 'good' system.

If mirroring is really such a good idea why do none of the competing products offer this capability.

Mostly because they can't, to implement mirroring it's essential to integrate it into the core mail server code at the design stage so they are too far down the path to add it.

Most other suppliers offer one of two alternatives instead, they either provide 'file system' level mirroring, which at best is much less efficient, and likely to be minutes or hours 'out of date'. Or they promote the 'shared network drive' approach even though this clearly fails to duplicate the 'data' and thus is completely ineffective as a fault redundant solution.

I have two live mail system that I would like to mirror each other. Can I and / how do I configure this?

Assuming these systems currently run different domains, yes you can, you first add the other domains to each of the servers, and then turn on the mirroring settings. Then issue a tellmail resync on 'both' of the servers so that each one sends the new domains to the 'other' system.

Can I check it's always working?

The best test is to send an email to one system, then read it from the other, you can setup our '[watchdog](#)' utility to do this automatically once an hour so that you will always know if anything goes wrong.

You can also check the mirror section of the 'status' page, here the cryptic errors are often not that important, the key thing to look for is the counters showing successfully mirrored items, are these counters ticking over.

Is there any functionality / settings etc. that are not mirror?

The best answer we can give here is 'probably'. :-). We've tried to identify everything, but there may be things we've missed. In particular if you add settings to your config file which refer to files that are non standard then those files may not get mirrored. (And alias file setting for a domain would be one example).

DLIST currently only runs on 'one' of the servers (the master), this is to avoid problems of mailing list messages being sent twice by mistake :-), It's files are mirrored though so the data is duplicated.

The user database is not mirrored unless you are using NWAUTH and you turn on the setting. If you are using some other user database then you will need to consider if it needs mirroring in some way. Usually in this

situation it won't be an issue as it will be a network accessed database anyway.

Please also note that the config mirroring is new and requires SurgeMail 3.1 or later for best results.

Any Performance Impact?

There is of course some and the data does need to be sent between the two systems. However, the load is by no means doubled, as the mirroring occurs at the delivery stage after much pre-processing has occurred (e.g.: spam & virus filtering). Also most mail servers will run at about 98% idle so the extra load is really of no relevance (even for quite large ISP operations). We run mirroring on the servers we host with 40,000 plus users on a system. So far we've had about 3 Raid/system failures on our own hosting systems where mirroring has 'saved the day' and resulted in no significant loss of data.

You can mirror over a WAN connection, but the round trip time may slow down the mirroring a bit so if the system is very heavily used it may struggle to keep in sync. On most systems this is not a problem. But on very large busy systems this would be a mistake.

Swap Master/Slave when a system dies or is replaced.

For short periods there is no need to swap master/slave. The only thing that doesn't function on the slave is mailing lists.

If the master is dead or is being replaced but it may take a week then you may choose to swap them, do it like this:

- 1) Stop both servers.
- 2) Change the `g_mirror_mode` setting from "slave" to "master"
- 3) If needed change/swap ip addresses for the servers.
- 4) start the new master server (which was the slave)
- 5) When the old master server is repaired be sure to set it's `g_mirror_mode` to "slave" before starting it!!!
- 6) Issue a 'resync_config' and 'resync' on the MASTER (which was the slave) once the new 'slave' is running.

Moving SurgeMail to a new machine or operating system

When upgrading hardware or changing operating systems you can use mirroring to move your installation without disrupting users.

1. Install surgemail on the 'new' system
2. [Setup mirroring settings on both systems](#), the existing system will be 'master' the new system will be 'slave'
3. On the master issue the commands:
 1. `tellmail resync_config`
 2. `tellmail resync_nwauth`
 3. `tellmail resync_mkdir`
 4. `tellmail resync`
4. Carefully check the new system is working and has all accounts/messages.
5. Move the license key
6. Move the users across.
7. Stop both servers and swap the master/slave settings so the new system is the master.

Performance and Scalability

SurgeMail has been designed from scratch with performance in mind. This means you can almost literally host "an unlimited number of users and virtual domains on a single system". For very large systems though you may want to use various of the proxy options below, first lets talk about hardware an OS.

[Technical info on clustering](#)

Hardware suggestions

Number of active users	Rough hardware outline
1-1000	Anything will work fine. We recommend you use a mirrored drive for the mail spool though.
1000-5000	Dual core Pentium SATA or SSD disk 2gig ram Install a SurgeMail/mirror system for failover.
5000-100000	4-8 processor core, I7 or Phenom II 8gig RAM Internal RAID 10 disk array (or SSD disk array) Seperate disk drive for operating system. Move 'SurgeWeb' to a seperate system. Install a SurgeMail/mirror system for failover.

Operating system options

Here we recommend you choose the operating system you are most familiar with from the choices of Linux or Windows , there is no significant performance difference between the two as far as SurgeMail is concerned so the key factor is which system you are best at doing the major common tasks on, e.g. OS install, configure, partition disks etc.

Scaling a system up from 100,000 to 100,000,000 users

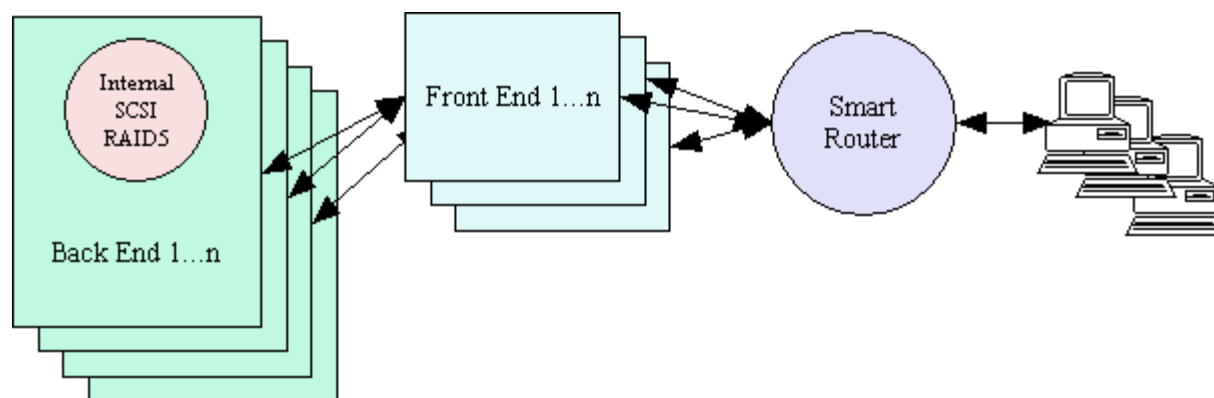
However we also had in mind ridiculously large systems with 1-100 million users, in these situations there are two methods you can use.

- Proxy Servers (huge systems)
- NFS / Shared Networked drives (huge systems)
- Simple splitting up of function (medium sized systems)

Proxy Servers (huge systems)

This systems allows both infinite scaling, and 3 layer security. The incoming POP/SMTP connections arrive at one of several front end 'proxy' servers (running SurgeMail in proxy mode) these servers then lookup the user in the networked user database (via LDAP or our own TCPAuth module) and along with the normal response an extra response code of 'tohost=backend.host.name' is returned, the proxy then redirects the user to the appropriate back end system. **Note: This mechanism doesn't currently support some web based user features (friends,exception rules etc)**

So you might run 4 back end systems, each with 100,000 users, and 2 front end systems. To add more users you just add as many front end and back end servers as needed to cope with the load.



Each user is only on one of the back end systems, the only piece in the system that has to handle all the users is the user database, which is a relatively trivial task as the quantity of data per entry is so small. We recommend the use of NWAAuth or LDAPAuth but any of the database back end authent modules would be suitable.

[See here for technical details](#)

Note: 3 Layer Security: This model is called '3 layer security' as the front and back end systems can be separated by another fire wall. And in the case of 'WebMail' the user web interface can also be separated from the front end systems by a fire wall, hence '3 layer' :-)

To implement this system set on the proxy system the setting `g_proxy true`, and in the authent module add the '`tohost=xxx`' field. For existing user accounts you can define `g_proxy_default host.name` so that user records with no '`tohost`' entry are correctly sent to the existing back end system. In this way a non proxy based system can be instantly turned into a proxy based system.

[Proxy config info](#)

NFS / Shared Drives, Clustering Support

In this mode you simply define your main drop path for a domain as a networked drive and setup multiple systems using that same drop path. As SurgeMail uses a 'maildir' directory format there are almost no locking issues even with bad NFS implementations (and most NFS implementations are a little dodgy :-).

[NFS/Shared drive config info](#)

Customising 'Look and Feel'

SurgeMails WebMail Web Look and Feel

WebMail is based on template technology which allows for very easy customisation of the look and feel of your web email service. There are many different ways of configuring WebMail to suit your needs but the most used and recommended methods of customising the WebMail look and feel with surgemail are listed below, with pros and cons :-). For some samples of the webmail template sets see <https://netwinsite.com/surgemail/templates.htm> and <http://netwinsite.com/webmail/gallery/index.htm>.

For customizing templates other than the WebMail templates (e.g. user self admin or SurgePlus templates) see <http://netwinsite.com/surgemail/help/templates.htm>.

1) "I just want simple WebMail rebranding" (global or per domain)

WebMail has a limited set of predefined variables that allow you to rebrand key WebMail functions without needing to modify the templates.

The following can be modified:

- WebMails display name and window title (cust_display_name, cust_title)
- Logo displayed in topleft of panel template window (cust_logo_url)
- Logout URL (cust_logout_url)
- Pane background colours and images (cust_panel_bgcolor, cust_panel_menu_bgcolor, cust_panel_fldbar_bgcolor, cust_panel_menu_background, cust_panel_fldbar_background)
- Display of custom links (cust_no_surgemail_links, cust_no_pgp_links, cust_no_admin_links)

This can be done on a per vdomain basis by adding these to vhost sections in surgehost.ini

eg:

```
vhost mymail.com
cust_display_name MyMail
cust_title MyMail
cust_logo_url /nwimg/mail/myimages/myimage.gif
vend
```

This can be done on a global basis using the WebMail manager interface. The global settings are stored in the file custom.dat and override any per domain settings.

Pros:

- You will not lose your rebranding when you install template updates

Cons:

- Rebranding is limited to the predefined rebranding settings

2) "I want different custom template set per vdomain"

If the above is not enough or you want a different template set for each vdomain this can be achieved by installing a different template set (images and template files) and setting the path to this template set on a per vdomain basis in webmail.ini as below. You will need to either copy the panel template set or download another template set from

<http://netwinsite.com/surgemail/templates.htm> (or a WebMail distribution)

```
vhost mydomain2.com
templates c:\surgemail\webmail\mydomain2.com
nwimg /nwimg/mail/mydomain2.com
vend
```

```
vhost mydomain3.com
templates E:\surgemail\webmail\mydomain3.com
nwimg /nwimg/mail/mydomain3.com
vend
```

Note: You should not have any global "tpl_set # ..." lines defined in this case as these global templates would be available for each vdomain. You can of course define multiple template sets for each defined vdomain using the following syntax:

```
vhost mydomain2.com
templates E:\surgemail\webmail\mydomain2.com
```

```
nwimg /nwimg/mail/mydomain2.com
tpl_set 1 E:\surgeemail\webmail\panel /nwimg/mail/panel Panel Set (Panel)
tpl_set 2 E:\surgeemail\webmail\marble /nwimg/mail/marble Marble Set (Marble)
tpl_set 3 E:\surgeemail\webmail\mydomain2.com /nwimg/mail/mydomain2.com Custom Set (Custom)
vend
```

Pros:

- You have complete control over the content and layout of your templates

Cons:.

- You will not get your templates upgraded to include fixes and new functionality by the SurgeMail installer

3) I want to customise the login page / simplify the login url for each domain

If you have a template set for each domain you can customise the login page by editing the login.tpl file.

Alternatively you can setup a login from your own arbitrary page by using a login based on the following HTML form. You would typically set this up on your main webserver or you could modify the surgeemail/web/index.htm file. (note: To make this work "no_tcode true" should be defined in webmail.ini)

```
<html><head><title>Some arbitrary page</title></head>
<body> WEBMAIL LOGIN
<form action="http://mydomain.com:7080/scripts/webmail.exe" method="post" name="login">
<input type="hidden" name="frames" value="true" />
User: <input type="text" name="user" value="" > (full username eg user@mydomain.com) <br>
Password: <input type="password" name="pass" value="" > <br>
<INPUT id=quick_login name=quick_login type=submit value=Login>
</form>
</body></html>
```

SurgeMail is distributed with one one templates set. Currently we are setting up a page for downloading a variety of [template sets](#) - if you have one you would like to share please let us know :-)

Alternative template sets can be downloaded for free as possible starting points for your own system. These are currently distributed as three template sets available with surgeemail. Further information (slightly historic) on customising WebMail can be found in the online version of the [WebMail manual](#) - Section [Templates](#).

SurgeMail Web Administration Look and Feel

SurgeMail web interface is based on the same flexible template technology. However there is normally less need for this.

Customising Protocol Prompts

The POP and SMTP welcome message is customisable per domain using give you is customisable using the settings: [POP welcome](#) and [SMTP welcome](#).

Customising Internal Email

There are several internally generated emails:

Delivery failure	Sent when a message cannot be delivered.
Delivery warning	Sent when there is a temporary delivery error.
Quota limit	Sent when a user reaches their quota.
Signup email	Sent to the user during the new user signup process.
Manager signup email	Sent to the manager during the new use signup process.
Account status	Sent to the user periodically.
Friends confirmation*	Sent to an unknown incoming email address for friends verification. (if in friends confirmation mode)
Kids-safe message	Sent to an unknown incoming email address. (if in friends kids-safe mode)
Centipaid message*	Sent when payment is required on an incoming message.
Email Notification*	Sent to a user supplied address to notify of incoming email on the account.

*These messages can be customised by the user as well, user customisations are stored in the users path and used in preference to defaults.

To customise a message you simply create a specific file in the SurgeMail installation path and into that file put the contents of the email. Some messages allow email headers to be included in the file, they are indicated below in the examples.

Type	File	Example	From header	MAIL FROM
Delivery failure	failed.eml	example	g_from_header	<>
Delivery warning	warning.eml	example	g_from_header	<>
Quota limit	quota.eml	example	g_from_header	<>
Signup email	signup_user.eml or .msg	example	g_from_header	<>
Manager signup email	signup_manager.eml or .msg	example	g_from_header	<>
Account status	status.eml or .msg	example	g_from_header	<>
Friends confirmation	confirm.eml* or .msg	example	"email (g_friends_name system)" <email>	email
Kids-safe message	kid-safe.eml* or .msg	example	"email (g_friends_name system)" <email>	<>
Centipaid message	centipaid.eml or .msg	example	g_from_header	<>
Email Notification	enotify.eml or .msg	example	enotify from, g_enotify from , or account email	<same as from>

*In addition to this file the text:

For more information see <http://netwinsite.com/surgemail/friends.htm>

or another file friend_footer.eml (or .msg) is appended to these messages. (was friend.eml in versions prior to 1.6h)

As there are several things that change for each instance of these emails there are several special text replacement macros you can use, these macros operate in the same fashion as the ones on the SurgeMail administrative and user web pages. In fact a lot of the macros will function i.e. `||localtime||` and `||ifdef||` etc...

In addition each message has it's own set of macros which are available, they are...

failed.eml

Macro	Content
domain	The domain name

Example

```
When trying to deliver your message, the mail server at ||domain|| encountered
problems with the following addresses.

For a more detailed explanation see http://netwinsite.com/surgemail/deliver_failed.htm
```

warning.eml

Macro	Content
domain	The domain name

Example

```
When trying to deliver your message, the mail server at ||domain|| encountered
problems with the following addresses. Still trying, no action required!

For a more detailed explanation see http://netwinsite.com/surgemail/deliver_warning.htm
```

quota.eml

May contain email headers at the start of the file, hard-coded headers include To, From, X-AutoResponder, and Subject (if there are no headers in the file).

Macro	Content
domain	The domain name.
reason	The reason for this message.
subject	Same as reason .
size	Size of the message that exceeded the quota (in bytes)
sizek	Size of the message that exceeded the quota (in kbytes)
max	Total size of quota (in bytes)
maxk	Total size of quota (in kbytes)
used	Amount of quota used (in bytes)
usedk	Amount of quota used (in kbytes)
usedandsize	Amount of quota used plus the size of the message (in bytes)
usedandsizek	Amount of quota used plus the size of the message (in kbytes)
user	The account username
sender	The sender of the message that triggered this warning

Example

```
Subject: Your quota has been exceeded used=||used|| msgsize=||size|| max=||max||
reason: ||reason||

Details: max=||max|| used=||used|| size=||size||

This message is an automatic warning that you need to reduce disk usage
you should read and delete messages from your inbox or other folders.

If your mail client is set to ignore messages over a certain size, or leave
messages on the server then changing that setting might be in order.
```

For a more detailed explanation see http://netwinsite.com/surgemail/quota_warning.htm

signup_user.eml

May contain email headers at the start of the file, hard-coded headers include To, From, and Subject (if there are no headers in the file).

Macro	Content
to	To address
from	From address
account	Account that has been requested
url_add	URL to accept/complete the account signup
url_decline	URL to decline/cancel the account signup

Example

```
We received a request for an email account '||account||'.
To create the account now please click the link below.
  ||url_add||
If you did not request that account or have changed your mind please click on the link below
  ||url_decline||
```

signup_manager.eml

May contain email headers at the start of the file hard-coded headers include To, From and Subject (if there are no headers in the file).

Macro	Content
to	To address.
from	From address.
account	Account that has been requested.
ip	Ip signup came from.
url_add	URL to allow the account signup.
url_decline	URL to decline the account signup.
url_block	URL to block future signups from the same IP.
begin_list_details end_list_details	This function loops X times where X is the number of fields the user filled in upon signup each time it loops it outputs field and value the field name and value the user supplied.

Example

```
The account '||account||' has been requested from ip '||ip||'
Details given:
||begin_list_details|| | | | |
  ||field||: ||value||
||end_list_details||

To ALLOW the account click the link below:
  ||url_add||

To DISALLOW the account click the link below:
  ||url_decline||

To BLOCK this signup and future signups from this ip click the link below:
  ||url_block||
```

status.eml

May contain email headers at the start of the file hard-coded headers include To, From, X-Friend and Subject (if there are no headers in the file).

Macro	Content
to	To address.
from	From address.
url	URL to user self administration web interface.
start	Time in seconds of last status message.
end	Time in seconds now.
friends	The g_friends_name setting.
begin_list_pending end_list_pending	This function loops X times where X is the number of emails in the users pending folder. Each time it loops it outputs pending_idx , pending_time , pending_name , pending_from , pending_subject , pending_score about the pending message.
begin_list_spam_store end_list_spam_store	This function loops X times where X is the number of emails in the users spam store folder. Each time it loops it outputs store_idx , store_time , store_name , store_from , store_subject , store_score about the stored message.
account_log	This function outputs the contents of the users account log
proto	Either "http" or "https" depending on server settings.
host	The host and port of the surgemail web interface.

Example

```
This message is an automatic status message sent to you because you are using ||friends||, Spam filtering,
Account exceptions, or CentiPaid on this email account.

To disable this login to the link below and go to the \"Log\" page.
||url||

Report for the period
||localtime||start||
till
||localtime||end||

||begin_list_pending||
||ifequal||pending_idx||0||
friends|| messages waiting to be confirmed:
endif||
pending_from||||pending_subject||
end_list_pending||

||begin_list_spam_store||
||ifequal||store_idx||0||
Held messages:
endif||
store_from||||store_subject||
end_list_spam_store||

Record of messages processed:
||account_log||
```

confirm.eml

May contain email headers at the start of the file hard coded headers include To, From, Subject and X-Confirm. The file [friend_footer.eml](#) (was friend.eml in versions prior to 1.6h) is appended to this message. See also [g_friends_confirm_subject](#).

Macro	Content
confirm	Unique message identifier (required in reply to this message to confirm original message release).
friends	The g_friends_name setting.
email	Person who is requesting confirmation (From address of this confirmation).
sender	Person who sent message that caused this confirmation (To address of this confirmation).
subject	Subject header from the message that caused this confirmation.
orig_email	To header from the message that caused this confirmation.

orig_sender	From header from the message that caused this confirmation.
-------------	---

Example

```
Your email to "||orig_email||" <||email||> with the subject "||subject||" has been blocked by the ||friends|| system. ||email|| only accepts email from people on their list of friends, you are not currently on their list of friends. If you would like to be added to the list simply reply to this message without changing the subject line.
```

kids-safe.eml

May contain email headers at the start of the file, hard coded headers include To, From, and Subject (if there are no headers in the file). The file [friend_footer.eml](#) (was friend.eml in versions prior to 1.6h) is appended to this message.

Macro	Content
friends	The g_friends_name setting.
email	Person who is denying message (From address of this confirmation).
sender	Person who sent message that caused this denial (To address of this confirmation).
subject	Subject header from the message that was denied.

Example

```
Your message with subject "||subject||" was rejected as "||email||" has not listed your email address "||sender||" as a known address. They are running in 'kids safe' mode so you must contact them via some other means to enable access.
```

centipaid.eml

May contain email headers at the start of the file hard coded headers include To, From, X-Autoresponder and Subject (if there are no headers in the file).

Macro	Content
amount	The amount being charged.
email	The person who received the message that caused this payment request.
link	Link to pay and release held message.

Example

```
Your email to ||email|| has been held-up due to its lack of postage. This email account requires that you pay a postage of ||amount|| to allow the email to be delivered. To make this payment, please click the link below: ||link|| When you complete the payment your original message will be delivered to ||email||.
```

enotify.eml

May contain email headers at the start of the file hard coded headers include Subject, To, and From (if there are no headers in the file).

Macro	Content
from	The address the original message was from.
to	The address the original message was to.
body	The first 1024 characters of the original message body (can be trimmed/shortened, see example below)

In addition any/all headers are available using ||, eg ||subject||

Example

```
Incoming message...
Subject: ||subject||
Date: ||date||
```

Message Reads...
||trim...(body,50)||

Language Translation

SurgeMail and user.cgi

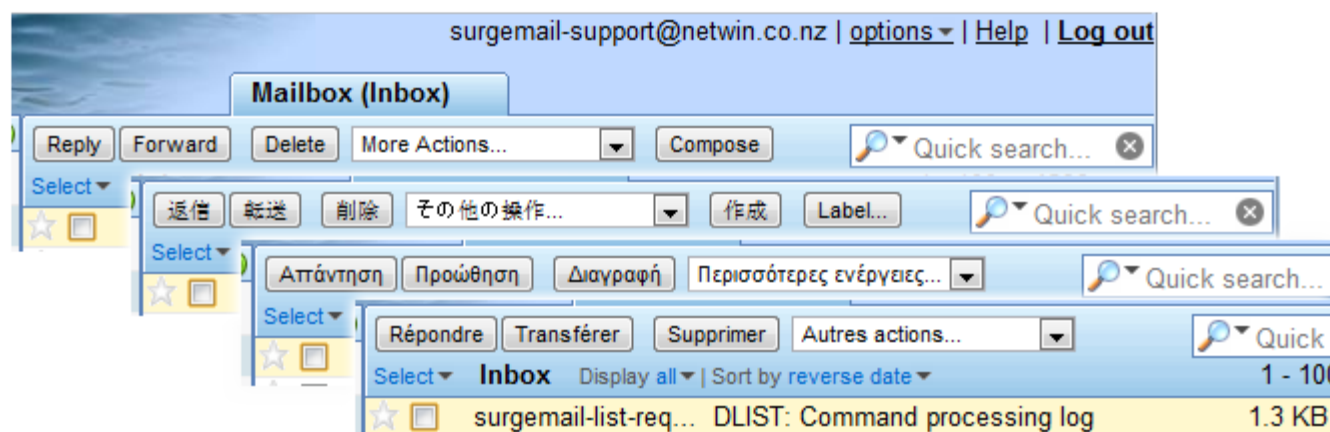
SurgeMail supports the translation of its web interface pages and internal messages via language translation files. These files contain a list of phrases and translations for those phrases in any number of languages. If you require a language which is not currently supported, or not supported completely it is simply a matter of editing the files and adding translations. We would appreciate it if customers would send us any translation files they do significant work on as we can then merge those additions into the default files so that other customers may benefit.

The SurgeMail files are called lang_web.dat and lang_bin.dat; lang_web.dat contains phrases which are found in the web admin and user self management pages (eg. c:\surgemail\web\), lang_bin.dat contains phrases which are found in the SurgeMail binary itself. In addition to these files SurgeMail will locate and use any file matching lang_web*.dat and lang_bin*.dat, this allows you to separate your languages out into files if desired, eg. lang_web_french.dat ...

To start using these files simply place them in the SurgeMail installation folder (eg. c:\surgemail) and restart SurgeMail. After changes to these files you will also need to restart SurgeMail. Any language SurgeMail finds in these files will be listed on the user self management login page. Selecting a language and logging in will store the selected language in a cookie on the users system. This cookie is shared by the WebMail interface, meaning that when the same user goes to the WebMail interface they will see it in the selected language. There are 2 places a user can change the selected language, the first is in the user self management interface, on the details page, the first page they see upon logging in. The next place is in the WebMail interface on the options page.

SurgeWeb webmail interface

The SurgeWeb interface has a separate language translation system, using separate files, though it is very similar. For details please go [here](#). Bear in mind that a few of the SurgeWeb pages use the SurgeMail self management system (user.cgi) to provide features such as Friends, Spam rules and Filtering so the phrases you are trying to translate may need to be translated in either set of files.



An significant feature of the SurgeWeb language customisation system is that it allows for arbitrary customisation of the English language phrases as well as actual translation for non English languages.

If you have any trouble translating any phrase please contact surgemail-support@netwinsite.com for help.

Mail Redirection - a guide

SurgeMail can redirect mail in one of four ways. These can be mixed but we suggest you pick one and use it as it can get very confusing:

1. Global [redirection](#) and [cc](#) rules
2. Per domain [redirection](#) and [cc](#) rules
3. Per domain [alias](#) files (for backwards compatibility)
4. Forward by returning new destination address from authent database

What is the difference?

- **Global redirection** takes place as soon as the message arrives and is applied to local and non local addresses. Global redirection rules should use fully specified email addresses.

[g_redirect](#) was="user1@fulldomain" to="user2@somedomain"

This is powerful but means you can unwittingly redirect outgoing mail. eg: redirection of postmaster@* -> admin@mydomain will redirect incoming mail for postmaster accounts on all local domains to admin@mydomain BUT will also redirect outgoing mail for postmaster@remotedomain to admin@mydomain!

- **Per Domain redirection** will be applied once a local destination domain is identified and allows individual accounts on a domain to be redirected. The match rule should not include the domain name:
[redirect](#) was="user1" to="user or user@otherdomain"
- **Alias files** for backward compatibility UNIX style alias files may be used to redirect mail for a certain accounts in a domain.
- **Fwd in Authent Database** If your authentication lookup returns a "fwd=new@address" field then the mail will be redirected to this account. This is done just prior to message delivery and after full processing of the message. This is the setting which may be changed by users using the web based account administration.

What happens if I different redirection mechanisms together?

Redirects under the same category will all be applied at the same time and multiple rules may be applied. Also redirection may be to accounts which are in turn being redirected.

1. Global redirection attempted. If [g_redirect](#) applied no further redirection is attempted (specifically this overrides #2 below)
 - [g_redirect](#)
 - [g_redirect_cc](#)
2. Domain based redirection. If [redirect](#) or alias file is applied no further redirection is attempted (overrides #3)
 - [redirect](#) = domain redirect
 - alias file redirection
 - [redirect_cc](#) = per domain cc
3. Authentication based fwd (overrides #4)
4. Message delivered normally to local account.

Advanced use of redirection

Both local and global redirection allow * wildcard to be used in the "was" field and the %n wildcard to be used in the "to" field.

eg. *_@domain -> %1@%2.domain

would allow joe_bloggs@domain to be redirected to joe@bloggs.domain

Gateways

The [g_gateway](#) setting is a final form of mail redirection that will will redirect mail for an entire domain (or subdomain) to a particular IP address.

Typically this other server is inside a fire wall so its local IP address is not known by the DNS server. You specify the domain and IP address (and optional login information) to send messages to and this server is treated as 'local' rather than remote in terms of open relay restrictions.

SurgeWall (Version 1.4a+ required)

SurgeWall is a special mode which SurgeMail can run it. In this mode it gateways email through itself to any existing mail server after first processing it with it's Friends, Spam, Virus and other mail rules and filters. One SurgeWall install can gateway email for any number of other servers; it is done on a per domain basis.

SurgeWall or g_gateway -- which setting should I use?

If you want to provide existing mail servers and their users with the SurgeMail user level spam, virus and friends features. Plus a web interface to allow users to configure these features then you want SurgeWall. SurgeWall will gateway mail to the backend server, pipe POP3 and IMAP connections there, and provide a web interface for users. It does not require SurgeMail to have access to the other mail servers user database, SurgeWall will use POP3 login connections to verify users.

If you simply want to gateway mail for domains to one or more backend servers without supplying the SurgeMail user level spam, virus and friends features then you should use the [g_gateway](#) setting. With g_gateway SurgeMail still does virus scanning, SPF, and some spam filtering, it is only the user level options which it does not provide. You can use [g_proxy to gateways](#) to get POP3 and IMAP connections proxied as well. There are a number of gateway configuration options, search for "gateway" in the web admin interface or the help.

Configuration

SurgeWall is enabled at domain level, this means you need to add a domain to the SurgeMail config for each domain you want to SurgeWall. The domain level setting is called '[surgewall](#)' you simply specify the address of the existing server. Example:

```
vdomain name="domain.com" address="1.2.3.4"
  surgewall "2.3.4.5"
```

The above setting will:

- Gateway all mail for users@domain.com to the server on 2.3.4.5, after applying SmiteSpam, friends, and the various configurable filters and virus scanners.
- Proxy any POP3 connections from 1.2.3.4 to 2.3.4.5.
- Proxy SMTP auth requests from 1.2.3.4 to 2.3.4.5.
- Verify accounts on 2.3.4.5 at the rcpt stage of mail delivery to 1.2.3.4.

Another configuration option is the [SurgeWall_options](#) setting this setting will contain all the miscellaneous SurgeWall options i.e.

```
surgewall_options strip_domain="TRUE" pop="" smtp="" imap=""
```

See the [SurgeWall_options](#) setting description for details about each parameter.

Smart Router / Load Balancer

If you're using a load balancer with SurgeWall then please read [this](#).

Name translation New (surgeemail 4.0v-9+)

You can now use surgewall on servers where the login name is different from the email address. To do so you create a file "surgewall_translation.txt" in the surgeemail directory that contains pairs of login names to be translated.

```
user@domain.com real_login_name
user2@domain.com real_login_name2
user3@domain2.com real_login_name3
```

Users login to user.cgi / webmail using their email address and password, but surgeemail will translate this as required just prior to connecting to the real mailserver.

note: This file is only loaded at startup so if new accounts are added to the configuration surgeemail should be restarted.

Extras

To enable you to configure a large number of domains quickly there is a 'tellmail' command called 'SurgeWall' which will take an

input file containing SurgeWall configurations and add them all to the SurgeMail config at once. It will simply append the data to the existing configuration, so running it twice will add 2 copies of each domain.. There is an example input file called 'surgewall_example.txt' in the SurgeMail installation folder. You can modify this file or create your own file. To use this command simply type "tellmail surgewall surgewall_example.txt" at command prompt.

SMS

Sending to an email to sms gateway

Lets assume the gateway you are sending to accepts sms messages of the form NUMBER@sms.com, then add the following settings to surgemail.ini

```
g_redirect was="msgate_*@*" to="%1@sms.com"
g_sms_gateway "127.0.0.1:2025" # this setting isn't used but is required to enable the sms pages.
vdomain name="x.y.z"
...
user_sms "true"
vdomain name="a.b.c"
...
user_sms "true"
```

NOTE: The rest of this page refers to the 'msgate' module which we no longer supply, it is recommended you use the above method instead for better/simpler results and lower cost.

The following documentation is only here for administrators already using msgate.

Simple notification of email via SMS

By setting the [g_sms_gateway](#) setting to the ip:port of the machine where SMSGate is installed, then setting [user_sms](#) to TRUE for each domain you allow your users to configure SMS notification of email based on rules of their choice. This means when an email arrives for this user that matches a rule they have specified the first ~160 characters of the email are sent to the cell phone number they provide. Example surgemail.ini settings:

```
global_settings
g_sms_gateway "127.0.0.1:2025"

vdomain name="domain.com" address=""
user_sms "TRUE"
```

The user can reply to the SMS notification specifying an email address in the reply their SMS will be delivered to the email specified.

Advanced SMS gatewaying

By configuring a [g_gateway](#) setting and a [g_redirect](#) rule, you can allow you users to send SMS to anyone simply by sending an email to a specified email account on your server. Example surgemail.ini settings:

```
global_settings
g_gateway domain="msgate.domain.com" to="127.0.0.1:2025" user="" pass="" relay="FALSE" check="" sms="TRUE"
g_redirect was="msgate_*@*" to="%1@msgate.domain.com"
```

Using the above configuration settings users can send email to msgate_<number>@domain.com and an sms containing the first ~160 characters of the email will be sent to <number>.

The relay="FALSE" parameter enforces the use of SMTP authentication, this ensures only authenticated users can use this feature.

The sms="TRUE" parameter instructs SurgeMail to send the users SMS number (if provided) to SMSGate which will then use that as the sending phone number meaning replies to the SMS will go to the users phone. (A user option will be added to enable/disable this feature per user in the near future).

Some SMSGate config settings are required for the above to work, they are:

```
smtp_prefix msgate_
accept_from *@domain.com
accept_ip 127.0.0.1
```

The above smtp_prefix setting matches the prefix shown in the g_redirect rule as the was="" parameter. The accept_from setting is a wildcard comma seperated list of valid email address to allow SMS sends for. The accept_ip setting specifies the ip address of the SurgeMail server.

Various SMS delivery options

SMSSGate has support for several different SMS delivery options including using a GSM modem, delivery via an online provider using HTTP or SMTP based requests and even untested SMPP support. For more information see [here](#).

Mailing lists and Bulletins - a guide

Mailing lists

SurgeMail is supplied with DList - a fully featured mailing list server. Mailing lists are administered via "Mailing Lists" on the main SurgeMail Web Admin interface.

Generally the Sysadmin would set up a list, and then users would send an email to the 'listname-request' address to 'subscribe' themselves to the list.

Users only interact with the list by sending emails either directly to the list to be 'posted' or to the listname-request address if they wish to join the list or send it commands. When users join the list they are normally sent this list of commands so that they know what the list can do for them.

The administration of mailing lists is fairly self explanatory, but if you run into difficulty or want to modify the dlist config files by hand here is the [detailed DList manual](#)

Users can create and manage their own mailing lists via the web user self management interface, see the [g_user_list_quota](#), or [user_list_quota](#) settings and also the [g_user_access](#) setting for how to enable/disable this feature.

Anonymous access to mailing list archives is available if the mailing list has it's web_archive setting set to true. The url is <http://localhost/list/<listname>> (where listname is the name of the mailing list that has a message archive). This interface allows searching of the messages and for people to subscribe and unsubscribe. The "join_cookie true" or "list_join moderator" is recommended to stop users signing other people up to mailing lists.

Bulletins

A bulletin is just like a mailing list message sent globally to all users or all users of a domain but much more efficient. A nice feature of bulletins is that they will be delivered upon account creation to accounts that do not yet exist.

SurgeMail / WebMail supports two kinds of bulletins.

SurgeMail bulletins can be sent from the domain page in the Web Admin interface and will be delivered to all users with an Email account on SurgeMail. Of note is that these bulletins will be delivered in the WebMail inbox as an email message.

WebMail bulletins are slightly different. These are sent using the WebMail Web Admin interface and will only be delivered to WebMail users as a message in the bulletins folder.

Securing the Server

- [SurgeMail SSL / TLS support](#)
- [Restricting Access by IP Number](#)
- [Relay restrictions](#)
- [Restricting mail services per user](#)
- [CRAM-MD5](#)

SurgeMail SSL / TLS support

SSL is fully supported on all protocols to ensure username and password are safely encrypted when sent over the internet so that they can't be stolen 'on the way past'. If you are running a mail server that doesn't support this feature then essentially anyone with access to your network can steal passwords. Almost all popular email clients now support SSL/TLS. Data is also encrypted, however be aware that when sending mail to other mail systems the data will be unencrypted on the journey, so only local Email is fully secure.

POP: Secure to regular port using STARTTLS, secure to dedicated port.

SMTP: Secure to regular port using STARTTLS

HTTPS: All web based administration tasks can be done either using secure HTTPS or standard HTTP.

Mirroring: The in-built server mirroring feature mirrors the server over a secure link.

SurgeMail SSL/TLS Frequently Asked Questions

- [What is SSL/TLS?](#)
- [How to generate a Certification Signing Request to get a CA signed key](#)

What is SSL/TLS and how secure is it?

SSL/TLS is the same encryption system used by 'https' web pages. It is generally considered to be the most secure method for sending sensitive information across the internet, and is the basis of most ECommerce security systems used today.

You will need a server private key (do not give this to anyone) and a matching Certificate which the server sends to the clients upon SSL handshake. The intermediate step is the Certificate Signing Request (CSR). This is generated from your private key and used to generate the certificate.

How to generate a Certification Signing Request to get a CA signed key

SurgeMail automatically generates untrusted certificates when required. For high level security you should consider getting your own trusted server certificate. This means that clients can be sure that they are talking to 'your' server and not just someone pretending to be your server and means that warning messages do not get displayed when connected using a browser or mail client attempting to use secure connections.

CSR generation is now built in to the surgemail. Simply go to the " SSL Certificates Configure" link on the globals page of the web admin. From here you can check the state of the current certificates for your domains and create a CSR and update SurgeMail with your signed key. Press the New CSR button to generate a new private key and matching CSR and untrusted certificate. Copy the CSR text and send this to your certification authority. You should be sent back a signed certificate to replace the automatically generated certificate - just paste this in the SSL Certificate(s) pane and press save. You will need to restart SurgeMail to get SurgeMail to use the new certificate.

Some certifying authorities issue trusted certificates based on a trust chain that involves an intermediate certificate. If you are required to install an intermediate certificate by your signing authority you can just place this in the surge_cert.pem file as follows (SurgeMail 1.5e+):

```
<begin surge_cert.pem file>
# Issued certificate for yourdomain.com
-----BEGIN CERTIFICATE-----
MIIFZjCCBE6gAwIBAgIQS288jS2Kir5dBc5Br8QmLTANBgkqhkiG9w0BAQUFADC
B
...
czNTZW1cm10eVnlcnZpY2VzXzIuY3J3MDggOKA2hjRodHRwOi8vY3J3LmNvbW9k
Q/az60ld5VnPDdz8kpNduHp4cWpVu9x3byRqWbm+UiaYRtANl/nhk9xx
-----END CERTIFICATE-----

# Certifying authority intermediate certificate(s)
-----BEGIN CERTIFICATE-----
```



```
MIIEYDCCBDGgAwIBAgIEAgACmzANBgkqhkiG9w0BAQUFADBFBMqswCQYDVQQGEwJV
...
BBYwFAYIKwYBBQUHAWEGCCsGAQUFBwMCMEYGA1UdIAQ/MD0wOwYMKwYBBAGyMQEC
vA2AOurM+5pX7XilNj1W6tHndMo0w8+xUengDA==
-----END CERTIFICATE-----
<end file>
```

Some certification authorities you could use are:

[DigiCert \(www.digicert.com\)](http://www.digicert.com/)
<http://www.comodogroup.com/>
<http://www.verisign.com/>
<http://www.thawte.com/>
<http://www.entrust.net/>
<http://www.e-certify.com/>
<http://www.digsigtrust.com/>
<http://www.globalsign.com/>
<http://www.tc-trustcenter.com/>
<http://www.valicert.com/>

As an alternative you can manually generate the the same files using the openssl binary (not distributed with SurgeMail):

```
openssl req -new -nodes -keyout surge_priv.pem -out surge_csr.pem
```

WARNING: If using GoDaddy certificates, be aware of a reported naming convention clash. To get your intermediate certificate you will want to download the "bundle" and not the "intermediate certificate".

Restricting Access by IP Number

Many of SurgeMails features can be restricted to certain IP number ranges. This can be used to make the system more secure.

One feature that is that should probably be restricted is the [g_admin_ip](#) setting to limit the valid IP addresses for SurgeMail server admin users.

You should also look into the following settings that control which connections will use SSL:

- [g_ssl_allow](#) - connections to allow to SSL use
- [g_ssl_require](#) - connections to allow to require SSL use
- [g_ssl_require_out](#) - outbound connections requiring SSL use

Relay restrictions

It is important to ensure that your system is not setup as an "open relay", as this is likely to result in spam being sent through your system and your mail server getting black listed by open relay databases.

SurgeMail "out of the box" is configured to not relay other than [allow relay after POP login](#) which, in general is safe and allows people using old mail clients (that do not know how to do SMTP authentication) to still send through your server without making your server an open relay.

A setting you may want to enable is [g_relay_allow_ip](#) for your mailserver's own IP address as this will enable other programs running on the system to send mail without needing to use SMTP authentication. Do not set this to * as this will make your system an open relay.

Other ways of enabling relaying is by destination domain ([g_relay_to](#)) or known from address ([g_relay_allow_from](#)).

Restricting mail services per user

Groups can be setup with rights to access POP, IMAP or SMTP services that will allow per user setting of access privileges. See [managing accounts](#) for more information.

CRAM-MD5

SurgeMail supports CRAM-MD5 SMTP authentication, but ONLY when using the NWAAuth authentication module. To enable

CRAM-MD5 set the [g_smtp_cram_enable](#) setting and restart SurgeMail.

This setting will cause NWAAuth to begin converting the stored passwords from their existing format into one that can be used for CRAM-MD5, as such users will have to login once to pop or imap before they can use CRAM-MD5.

WARNING: The stored CRAM-MD5 passwords are not as secure as NWAAuth's default SSHA passwords, they are only marginally more secure than plain text. A better solution to password security is to use [SSL / TLS](#).

Certificate file format conversion

This has not been explicitly tested but I header from a customer you can convert pfx certificates to pem using the following openssl command:

```
openssl pkcs12 -in {something}.pfx -out {something}.pem -nodes
```

If the certificate is protected by a password, you will be prompted for the password. Enter your password and the export is done.

WebDav Support

If enabled each email user gains access to a WebDav area on the server, this can be used to:

- Share files
- Publish small static web sites
- Generic network storage/backup
- Import/export calenders from some applications (e.g. thunderbird)
- You can mount this network area as a drive in windows (using 'webdrive' software)
- Use for iphone applications etc that require a webdav area.
- (Note webdav is NOT caldav, so it does not currently support specialized calendar sharing operations)

How to enable WebDav support:

- Upgrade to the latest release (4.2e-25 or later)
- Add these settings to surgemail.ini
 - **g_webdav_enable "true"**
 - **g_webdav_path "c:\surgemail\webdav"**
 - **g_webdav_public "true"**

How the user accesses their webdav directory

Use the following urls, and the user will be prompted for their normal email password, their email quota will also apply to their webdav folders!

- **http://your.server.com/wd/username**
- **https://your.server.com/wd/username**

A 'port' may be needed if you have the webmail port defined as 7080, e.g.

http://your.server.com/wd/username

If the g_webdav_public setting is enabled then your PUBLIC files can be stored in a directory

http://your.server.com/wd/username/pub

and this will be visible to anyone via a url of this form:

http://your.server.com/wd/username/pub/myfile.htm

Note: this only allows the display of static pages, it doesn't support cgi/perl or any other server scripting. (It does permit browser based scripting, e.g. javascript)

If you wish to control access to this feature on a user/domain level then define **"g_webdav_group "true"** then define the domain level setting: **access_group_default "webdav"** or add userdb field **mailaccess="webdav"** Which allows you to enable webdav on a user or domain based level.

Applications that can make use of webdav. Most are not free!

- (Tested) WebDrive (allows you to map a drive in windows)
- (Tested,Free) Windows built in WebDav mini-redirector ([see notes](#) and warnings)
- (Tested,Free) Thunderbird (allows you to send your calendar file to a shared area)

Super Flexible File Synchronizer 4.76 (shareware)

- GoodSync (backup utility) <http://www.goodsync.com/>
- (Tested) iPhone DAV-E
- (many others, please let me know good examples you come across)

Windows WebDav mini redirector / mapping a drive, bugs, issues...

To put it mildly, the built in windows webdav client is a piece of junk :-). But it can be made to work, try the following magic incantations:

- Install the hot patch from microsoft if applicable to your system. <http://support.microsoft.com/?kbid=942392>
- Connect by selecting 'map drive' then use the HTTPS address of your server
- Ensure surgemail is listening on port 443 so no port is needed.
- You may need to apply this <http://support.microsoft.com/kb/841215>

Or if you would rather not have the hassle, pay the \$40 for webdrive :-)

Some known bugs with the windows redirector are listed here:

<http://www.greenbytes.de/tech/webdav/webdav-redirector-list.html>

CalDAV Calendaring

This provides surgemail standalone CalDAV calendaring support (including calendar sharing) for mobile devices and desktop clients.

CalDAV Quick Start for administrators

1. Install from the admin interface and test the environment is working correctly:
 - [Install](#) using admin web interface - value added features - calendar sync - install
 - (Unix only) verify system php-cgi is installed (version 5.3+) and install if needed
 - (Unix only) point g_web_php_exe at system php-cgi (version 5.3+)
 - [Verify](#) environment is running
2. Login with your calendaring [client of choice](#):
 - Surgeweb : make sure this is enabled in admin interface if needed
 - iOS : Should autodetect caldav url:
Just specify server, username (eg user@domain.com), password.
 - Other clients: You may need to manually enter the calendar url:
caldav url: yourserver.com/cal/principals/user@domain.com
username: user@domain.com password: {your password}
 - Tested to work with iOS, OSX iCal, Mozilla Lightning, Android CalDAV Sync
3. Customise [sharing](#) of calendars using surgeweb:
 - Surgeweb - calendars - left column - configure caldav calendars

Installation Notes

Background

CalDAV calendaring support is based on the **SabreDAV** php library and is hosted by the surgemail webserver. This is integrated with surgemail for authentication. So any user with a valid surgemail email account should also be able to use CalDAV calendaring as soon as this has been enabled and installed

SabreDAV is **php** based. As surgemail does not use php technology elsewhere a suitable php environment needs to be installed. On windows the installer will setup a fully self contained and functional php-cgi installed in the surgemail {g_home}/php directory. On unix systems you will need to make sure you have a suitable php-cgi already installed (version 5.3+).

SabreDAV itself is a set of php scripts. Additions have been made to integrate with the surgemail for authentication and to **share calendars** with other users on the same server. This sharing is configured in surgeweb, and individual calendars may be shared as **read/write**, **read only**, or **free/busy** only. Actual calendar access is not yet available in surgeweb.

Installation

Installation of the calendaring php scripts can be done from **admin interface** (value added features - calendar sync) or using "**tellmail caldav_install**".

- On windows this installer also installs a standalone php-cgi config in surgemail/php.
- On unix systems you will also have to make sure there is a system installed fully functional php-cgi (version 5.3+) and check g_web_php_exe is correctly set to use this.

Running the installer, should result in the following output and a running calendars implementation.

```
C:\surgemail>tellmail caldav_install
```

```
CALDAV INSTALLER - this will enable surgemail hosted CalDAV calendaring
```

```
This installer will update :
```

- Standalone php-cgi distribution (surgemail/php/*) [windows only]
- SabreDAV and NetWin SabreDAV extensions (surgemail/phplib/*)

CalDAV Calendaring.

- Calendar script, test scripts, calendar data (surgemail/scripts/*)
- Any necessary surgemail.ini settings
(any existing surgemail CalDAV calendar database will remain intact)

Note: CalDAV provides standalone calendaring for mobile and desktop clients and is currently NOT YET integrated with the surgeplus calendar or surgeweb.

```
-----  
Downloading php distribution [php_windows.zip]  
Downloading [php_windows.zip] (52.09% of 7,368 KB)  
Downloading [php_windows.zip] (100.00% of 7,368 KB)  
Distribution ready for installation [php_windows.zip] size=7545115  
PHP distribution [php_windows.zip] extracted and installed  
  
Downloading php libraries (SabreDAV and NetWin extensions) [phplib.zip]  
Downloading [phplib.zip] (36.49% of 467 KB)  
Downloading [phplib.zip] (100.00% of 467 KB)  
Distribution ready for installation [phplib.zip] size=478729  
PHP libraries (SabreDAV and NetWin extensions) extracted and installed
```

```
Installing surgemail integration files  
Directory created [c:\surgemail\scripts\data]  
Installed [c:\surgemail\scripts\phpinfo.php]  
Installed [c:\surgemail\scripts\netwin.php]  
Installed [c:\surgemail\scripts\cal.php]  
Installed empty database [c:\surgemail\scripts\data\caldb.sqlite]
```

```
Verifying surgemail.ini settings  
added: g_url_redirect from="/.well-known/caldav" to="/cal"  
added: g_url_alias from="/cal" to="/scripts/cal.php"
```

Installation complete, and should hopefully be "ready to use" for your surgemail users :-)

Now now test this yourself to confirm all is well:

- Verify base PHP installation: <http://yourserver/scripts/phpinfo.php>
- Verify configuration for SabreDAV: <http://yourserver/scripts/netwin.php>
- Verify authentication integration: <http://yourserver/cal>
- Connected using calDAV client eg iOS device

For further info see <http://netwinsite.com/surgemail/help/caldav.htm>

C:\surgemail>

Verify system php-cgi (Unix / OSX only)

Many unix distributions differ a little when it comes to installing and configuring php (both in terms of package names and in terms of installer utilities). But the following instructions possibly with minor modification as a result of googling install instructions on your distribution of choice should get you a long way. The following works under recent Ubuntu linux:

```
root@svr:/usr/local/surgemail# which php-cgi  
root@svr:/usr/local/surgemail#
```

Run above, shows: "bother no php-cgi installed", so run :

```
apt-get install php5-cgi  
...
```

After successful installation:

```
root@svr:/usr/local/surgemail# which php-cgi  
/usr/bin/php-cgi  
  
root@svr:/usr/local/surgemail# php-cgi -v  
PHP 5.3.5-1ubuntu7.10 with Suhosin-Patch (cgi-fcgi) (built: Jun 19 2012 00:54:37)  
Copyright (c) 1997-2009 The PHP Group  
Zend Engine v2.3.0, Copyright (c) 1998-2010 Zend Technologies  
  
root@svr:/usr/local/surgemail#
```

Great have php set to run, use surgemail.ini setting of and restart surgemail or issue "tellmail reload".

```
g_web_php_exe "/usr/bin/php-cgi"
```

Now run the web based verification as documented in the next step and install additional packages as needed shown below. eg:

```
apt-get install php5-imap  
apt-get install php5-sqlite
```

Verify the environment

There is quite a lot that "may not be working" after the above so there is a simple three step verification proces to go though.

1. Verify php is running

The script `surgemail/scripts/phpinfo.php` script is a minimal php script to see whether surgemail is able to correctly run php scripts using php-cgi. Connect to this by browsing to the url:

`http://yourserver.com/scripts/phpinfo.php`



Correctly working php-cgi

If this request fails php-cgi is not setup correctly. Make sure `g_web_php_exe` is correctly set and check your php configuration and log files for possible sources of the fault. On windows these can be found `surgemail/php/php.ini` and `surgemail/php_errors.log`. On unix it will be system specific.

2. Verify php config for sabredav

In order to run CalDAV calendaring php needs to be at least version 5.3+ and have the modules PDO, PDO_SQLITE and IMAP installed. This can be verified using the url:

`http://yourserver.com/scripts/netwin.php`



CalDAV php-cgi prerequisites met

Any missing modules should get noted on this page. If modules are missing or php version is older than 5.3 you will need to upgrade your php installation.

This should be unnecessary and all setup on windows, but on unix you may need to upgrade your installed php version or install additional modules.

3. Verify authentication integration

Lastly verify the authentication. By default calDAV will connect to surgemail imap on `127.0.0.1:143` to verify the authentication information. To test this is working use a browser to connect to the SabreDAV debugging interface, and login with your full email address "`user@domain.com`" and password:

`http://yourserver.com/cal`



Authentication worked

On some servers, particularly older servers it may be a little tricky meeting these prerequisites. Notably php-cgi 5.3, PDO and PDO_SQLITE are essential. The imap module is recommended but optional. It is recommended you use imap based authentication, but if for some reason that is not possible you can edit the `surgemail/phplib/netwin/Nwauth_sabre.php` to authenticate by connecting to `nwauth` directly in which case you will not need the `imap` module installed in `php-cgi`.

If there are any issues first check the `surgemail/scripts/cal.log` file and then the `php` log file to try and identify the source of the issue.

Now on to the [client configuration](#).

How to configure incoming MX/SMTP servers

In some situations you will want 1 or more incoming smtp servers, there are some settings you can use to optimize performance in this situation:

Backend server ip address 1.1.1.1

Incoming MX server address 2.2.2.2

On the backend system (1.1.1.1) set:

```
g_spam_allow "2.2.2.2" (Version 3.0 and earlier)
g_gateway_allow "2.2.2.2" (Version 3.1 and later)
g_spf_skip "2.2.2.2" (Version 3.0 and earlier)
g_received_name "your.domain"
```

On the incoming MX server set:

```
g_received_name "your.domain"
g_vanish_bad_bounces "true"
g_send_max "40"
g_send_nolimit "true" (version 3.1 and later)
g_send_max_perdom "20" (Version 3.0 and earlier)
```

Smart Router / Load balancing

When you use a Smart Router or Load Balancer incoming connections are distributed amongst your servers, this can cause WebMail options to fail with 'auto-login' errors. If you have this problem then the solution is either to upgrade to a newer version of SurgeMail and WebMail, to bypass the load balancer, or to add [webmail_host](#) settings to each domain affected (which is usually all of them).

If you are running webmail on the surgemail machines then you need to upgrade or add webmail_host. The setting required is:

[webmail_host](#) "127.0.0.1"

The reason for this is due to the way in which WebMail handles those options, they are actually supplied by the user.cgi handled internally by SurgeMail. WebMail passes the users authentication information to SurgeMail then redirects the user to it. The problem that occurs with a load balancer is that the connection from WebMail to the server via the load balancer will likely go to a different server than the one the user is later redirected to.

This is the case even when the load balancer is smart and sends connections from the same ip to the same place because WebMail's ip is not the same as the users ip. The solution is to stop WebMail connecting to SurgeMail via the load balancer but rather to connect to the server upon which it is hosted, the same server that the load balancer will direct the user to (assuming it has the same ip to same place feature enabled).

The setting [webmail_host](#) is required only on earlier versions of SurgeMail (3.1e-3 or earlier) because in these versions the domain A record name was being used as the address to connect to, this A record name is typically set to the external name/ip of the load balancer.

If you are running WebMail on a seperate machine to SurgeMail you cannot have it connect to SurgeMail via the load balancer, auto-logins will fail. The reason they will fail is that the webmail connections will come from the ip of that machine but the autologin request will come directly from the user, so the ip's will not match and it is unlikely that the load balancer will select the correct surgemail machine (the same one as webmail was using for that user).

To enable auto-logins to function in the above situation you have to bypass the load balancer. To do that configure webmail to connect directly to a single backend surgemail server and also configure the url section of the netwin_autologin_id settings such that they are complete urls directly to the same backend surgemail server bypassing the load balancer.

NetWin Database Mode

Enable using: `g_maildir_netwin "true"`.

- [How it works](#)
- [What advantages does it have](#)
- [What disadvantages does it have](#)
- [How do I turn it on](#)
- [Utility commands](#)

Warning! This feature is in Beta test stage, we have run it ourselves for several months and have had no problems. We haven't heard from any customers about any problems either. We would definitely recommend keeping a backup(as you should anyway) if you decide to go with this method or running a mirror that is using the old/standard maildir method.

How it works:

Instead of storing each mail message as an individual file messages are stored in 'bucket' files which contain many messages, an index file contains a list of the contents of all of the bucket files.

So a folder with 100 mail messages in it might look like this:

```
28/09/2005 05:15 p.m. 1,120 bkt.idx
28/09/2005 05:15 p.m. 0 bkt.lock
28/09/2005 05:15 p.m. 7,578 bkt_0.bkt
28/09/2005 05:15 p.m. 4,450 bkt_0.head
26/09/2005 05:09 p.m. 442,871 bkt_1127711348.bkt
26/09/2005 05:08 p.m. 31,668 bkt_1127711348.head
```

When a bucket file is partly empty due to deleted messages it is re-written.

New messages always go to bucket zero.

What advantages does it have

- When accessing lots of little files the disk IO is greatly reduced, and the speed can thus be 2-4 times faster.
- IMAP gets the most noticable speed improvement.
- When backing up a mail partition the operating system has a much simpler job, this can make backup runs 4-14 times faster.
- Some operating systems either fail completely, or run very badly when a disk has millions of files on it, this reduces this problem drastically.
- If sharing a drive using NFS it should work as long as fcntl locking works on your NFS drives.

What disadvantages does it have

- It's a new feature which involved massive internal changes to surgemail, there is significant risk that bugs could exist in it.
- It is a lot more complex than the 'file per message' model, so is more prone to bugs.
- You can't manually examine a users mail box and modify files (although there are some commands to let you do this to some extent)
- Message delivery is about 20% slower! (as the indexes are written at delivery time)

How do I turn it on ?

1. Make a backup of your mail spool (or mirror it)
2. Set the new setting `g_maildir_netwin "true"`
3. Restart surgemail
4. An automatic conversion process will start, in addition any mailbox which is accessed will be automatically converted.
5. (You can turn it off the same way, change the setting and restart surgemail)
6. DO NOT DOWNGRADE TO AN EARLIER VERSION OF SURGEMAIL AFTER CONVERTING :-)
7. You can turn it on, on ONE member of a mirrored pair of surgemail servers.

Utility commands.

Command	Decription
tellmail dir user@domain [folder]	Show files in that inbox/folder
tellmail ndb_export user@domain foldername destination_path	Export files to a directory
tellmail ndb_repair user@domain.name	Repair index based on bucket files
tellmail ndb_convert	Start convert process again

Guide to using DomainKeys with SurgeMail (3.7c)

- [How it works](#)
- [What you need to do to enable DomainKeys checks for 'incoming' email](#)
- [What you need to do to generate DomainKeys signatures for 'outgoing' mail](#)

How it works:

DomainKeys is a cryptographic method that allows a receiving server/client to verify that the From/Sender header was accurate and not forged.

It does this by looking up the senders _domainkey.your.domain dns record to get the public key which it uses to check the signature in the message headers is correct.

SurgeMail makes use of this information to avoid grey bouncing a message when no SPF information exists. And may in future score signed messages differently.

SurgeMail can also 'sign' outgoing email, this helps your email get delivered to servers that use this information to further verify a message. And this makes it harder for spammers to forge your domain successfully.

There is a button in surgemail to generate your private/public keys. This creates the file domainkey.pem, if you have several servers sending email for your domain you will need to copy this file to each server.

As well as entering your public key into your dns you will define your policy in the txt dns record default._domainkey.your.domain and _domainkey.your.domain, this policy defines if you are testing or not, and if you sign all or some of the messages from your domain. A receiving system 'should' use this information to determine what action is valid if a signature does not exist or fails to verify.

What you need to do to enable DomainKeys checks for 'incoming' email

1. Upgrade to SurgeMail 3.7c-25 or later.
2. In the web admin tool, goto the DomainKeys page (spam_control - Alternative sender verification)
3. Turn on these settings:
Check incoming DomainKeys signatures g_domainkeys_check [TICK]

What you need to do to generate DomainKeys signatures for 'outgoing' mail

1. In the web admin tool, goto the DomainKeys page
2. Turn on the setting
Check incoming DomainKeys signatures g_domainkeys_sign [TICK]
3. Press the "Configure" Domainkeys button to generate your keypair and fetch your public key
4. Enter your public key into your dns server in the appropriate txt record as described on the page. e.g. default._domainkey.your.domain

SurgeMail IPV6

From 4.0t-5 onwards surgemail has support for IPV6 dual stack systems on some platforms. It will work correctly in a combined ipv4 ipv6 network. To enable it you will need the following:

- Upgrade to SurgeMail 4.0t-5 or later on a supported platform (Currently Windows Vista or Linux)
- Check for "IPV6 supported" in the advanced status page (at the bottom).
- Set in surgemail.ini **g_ipv6_enable "true"** and **RESTART SURGEMAIL**
- You will need an IPV6 address for your computer!
- If you wish to accept incoming IPV6 connections you must add AAAA records to your DNS entry for your mail server host.
- In tellmail status output you will see the new long addresses for connections that have come from IPV6 hosts.

Note: Some features are not currently implemented (spf, known ips etc), we will add these when need arises.



SurgeVault Encrypt HIPAA compliant feature

The SurgeVault feature allows you to define some rules (per domain) that specify when a message should be encrypted (based on subject, or content or destination etc) and then instead of sending the raw naked message the destination user is either sent an encrypted message, or a link to an encrypted message. In either case the destination user is required to login and set a password to read that and future messages. Then they are either shown the message, or given a key to decrypt the message they were sent.

How to configure/turn on SurgeVault encryption:

1. Upgrade to SurgeMail 4.2b-11 or later
2. Set the global setting **G_ENCRYPT_SURGEWEB_SHOW "true"** if you want the encryption icon at the top of surgeweb compose new email page to appear.
3. Set a domain level rule in surgemail.ini for each domain you want to be able to send encrypted messages (without this you can only send encryption from surgeweb)
encrypt_rule header="subject" contains="encrypt:" method="server"
4. Send an email to someone from the domain in question, with "encrypt:" in the subject.
5. Or, in surgeweb send an email to someone and click on the encrypt icon before sending it.
6. If you wish to use the feature regularly you will need a new Key to enable this feature (sorry this is a paid add on feature), otherwise it is limited to '2' messages per day!

Inline based encryption

In this mode the message is encrypted, then sent to the destination user as an html attachment which contains javascript to 'decrypt' the message, to obtain the 'key' to decrypt the message the user must login to the sending server and request it. The first time they do this they must set a password. This means the security of 'subsequent' messages is enhanced as the password cannot be 'reset' by the receiving customer. (this applies to the server based method too)

Server based encryption

In this mode the destination user is sent a link containing a key that is needed to decode the message which is kept on the sending server. This is equally secure.

Secure Reply

In either case a secure reply can be sent once the user has logged in to fetch the key or decrypt the message.

Encoding used

AES 256 CBC mode with MD5 hash.

How secure is it - what does it protect and what doesn't it protect you from...

After the first email exchange and the password for a user has been set, then the encryption will prevent someone spying on the message in the 'middle' between your sending sever and the receiving user. It does not prevent the administrator of your server from spying on the message as they can certainly circumvent this mechanism (with some difficulty).

However it provides you with a way of being sure that no one outside your server see's the message other than the intended recipient and it also gives you an audit trail to know that the receiving user did (or didn't) view the message. You can further enhance security by using https and ssl to send the message so that no one other than the administrator on your network can spy on the message before it gets to your server.

This mechanism is suitable and possibly a legal requirement for some forms of email, for example when a doctor sends an email to a patient that includes test results it would be an appropriate way of doing it. Or any time someone is sending personal private information via email and must provide some assurance that the message cannot be intercepted trivially!

Relevant Settings

Setting	Description
G_ENCRYPT_EXPIRE "30"	Days to keep encrypted messages before deleting
Domain based settings	
encrypt_rule header="subject" contains="secret" method="server"	Specify rule for encrypting messages
encrypt_subject	Private message
encrypt_intro	Please click on the attached message to read your secret message

Full encrypt_rule settings are:

encrypt_rule header=string contains=string from=string to=string method=string

SurgeWeb integration

In addition to encrypt_rule rule based triggering, the sending of encrypted email is integrated into the surgeweb compose pane.

```
g_encrypt_surgeweb_show true
```

Also note that there is a setting on the surgeweb customisation page that disables the SurgeVault interface in surgeweb.

```
encrypt_hide true
```

Warning replying to messages:

If you use a rule like this:

```
encrypt_rule header="subject" contains="encrypt:" from="" to="" noconfirm="" method="server"
```

And you send someone an email, lets say for some reason they cannot read it and send you a reply then you reply to their email the 'encrypt' rule will still match and the message will be encrypted again... So just be aware of that! :-)

Obviously this is normally exactly what you want so all your emails to them on this subject remain encrypted.

Configuration Examples

Please email surgemail-support@netwinsite.com any suggested examples you think we should add to this page.

- [Accept email for a private domain name "private.name" as if it was sent to your real domain name "company.com"](#)
- [Rewriting the 'from' envelope and from/sender/reply-to headers](#)

Accept email for a private domain name...

1. Create a domain called company.com
2. Go to the 'Alternate Names for this Domain' and click the Edit Rules
3. Add 'private.name' (or *.private.name)
4. Restart?

Rewriting the 'from' envelope and from/sender/reply-to headers

In surgemail.ini add: (using SurgeMail 3.1e4 or later)

g_from_rewrite was="*@privatename" to="%1@public.domain.com"

In mfilter.rul add:

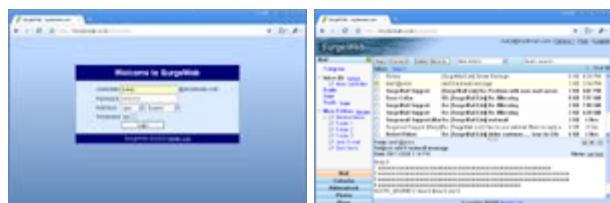
```
call replace("from", "@privatename", "%1@public.domain.com")
call replace("reply-to", "@privatename", "%1@public.domain.com")
call replace("sender", "@privatename", "%1@public.domain.com")
```

SurgeWeb Introduction

SurgeWeb is the high performance web based user interface for surgemail users. It's core design goals are to provide a modern web email interface that is:

responsive, efficient, customisable/extensible, maintainable

This page describes some of the things you should know about surgeWeb, particularly compared to the existing netwin webmail. For starters this is what the default interface looks like:



Login page

Main interface

Why is it better? Here are some key features that may convince you:

Key user features

- Much faster and more responsive
- Address autocompletion
- 'Drag and drop' message moving and copying
- Process multiple concurrent messages using tabs or popup windows
- Allows sending of HTML formatted messages from any browser
- Messages are automatically saved as drafts in background
- Keyboard shortcuts for most email processing operations
- Automatic image downsizing when emailing large images
- Attachment upload in the background while continuing to edit email
- Audio notification when new mail arrives
- Support for collapsible, nested folders
- Improved support for display of HTML messages
- Ability to mark messages as unread
- Relogin to timed out sessions without losing messages you are currently editing
- Fast indexed searching based on message subject / to / from
- Labels support for flexible tagging and organisation of messages

Key administrator features

- Support for multiple sessions on the same account
- Easy to customise
- Customisable at the global, domain or surgemail user group level
- Lower bandwidth usage
- Proper message redirect, as well as forward and forward(attach)
- Ability to add advertising banner
- Extend and customise without modifying core interface templates - much easier upgrades!
- Shared addressbooks, and external LDAP or surgemail user database addressbook support
- Proper support for international character sets (multiple concurrently)
- Easy internationalisation of the interface (sample translations of 26 languages)
- Support for separate English UK and English US

And what else?

SurgeWeb is setup with three interfaces:

Ajax: Fast responsive for faster computers - particularly on higher latency network connections

Basic: A traditional HTML interface for older computers

Mobile: An XHTML interface for small screen browsers

The **ajax interface** is the primary interface and uses modern web2 / ajax techniques to do much of the communication with the server in the background so that you do not have to wait for the server to respond to each request. You do need a relatively modern computer (approx less than 5 years old) to make effective use of the Ajax interface or it will appear sluggish as the client side processing will be slower than the server roundtrip delays. In the case of older computers it is recommended that the Basic (HTML) interface be used. The surgeweb Basic interface is much faster than the equivalent old Webmail HTML templates.

The use of either SurgeWeb interface uses **less bandwidth** than the old webmail templates. In addition to this http compression is used on all requests to the server reducing the bandwidth usage even further. The Ajax interface does have a slightly higher initial bandwidth requirement than Basic interface, and Basic interface does have a slightly higher ongoing bandwidth requirement than the Ajax interface.

There are also some nice features to make use of higher bandwidth if it is available. One of these is **inbox caching**. SurgeWeb will download the first page of messages in the inbox and cache these client side for instant display when when a user selects a message. This can be optionally disabled (useful for people on modem lines) or messages in other folders can be optionally cached for instant display. This means the end user experience when dealing with messages that are cached is as responsive / if not more so than a normal desktop email client.

In addition to the design goals mentioned above, work on SurgeWeb has concentrated on making SurgeWeb a **reliable and maintainable** email client - most importantly when it comes to HTML messages. We have plans for further integrating the surgeplus and user cgi features, but are not yet sure on the final form that those will take. For now these are accessible as pages within an iframe in the SurgeWeb interface.

Work in progress

SurgeWeb does remain a "work in progress", with many features planned and on the wish list but still to be implemented. For more information see the [buglist / feature request page](#).

Anyway, it must be about time for more serious feedback from the surgemail user community. Please let us know the things you do like, the things you don't like, and suggestions for how SurgeWeb can be improved and of course any outright bugs you find.

The NetWin team :-)

SurgeWeb FAQ

Do I need surgemail for surgeweb?

Yes and no, for performance reasons Surgeweb is run as an integral part of Surgemail yes, however surgeweb can be configured to talk to a separate [backend IMAP mailservers](#) - either surgemail or non surgemail.

Can I run surgeweb on a separate server?

Yes, either using a split [frontend / backend split](#) configuration to talk to backend surgemail servers, or using surgemail in surgewall mode to use any [existing imap server](#) backend.

Why is surgeweb better than webmail?

[Lots of reasons](#). Surgeweb uses modern AJAX / Web2 techniques to provide a web based email client that is as responsive and usable, if not more so, than a normal desktop mail client.

Does surgeweb have shared address books?

Yes, surgeweb has a flexible facility for [shared contacts](#) - at the global, domain or group level.

Can I run surgeweb against a POP only server?

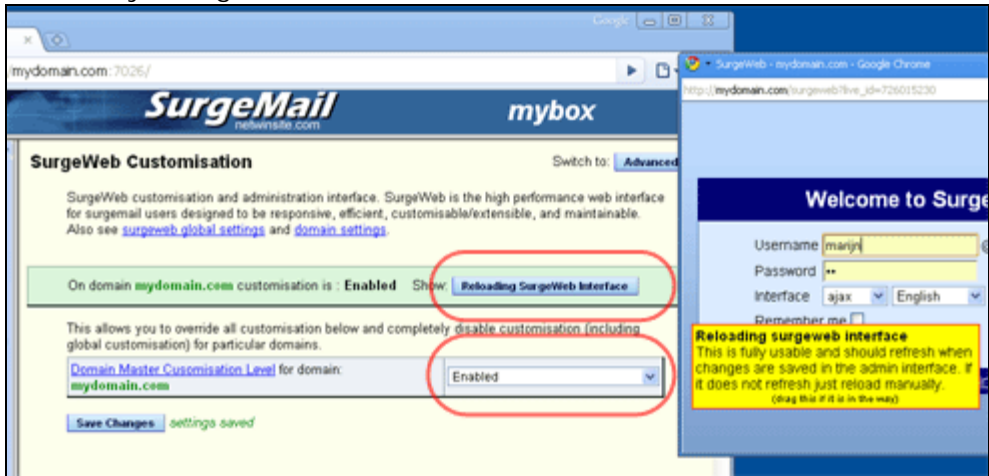
No, Surgeweb needs IMAP support and will not work with POP only servers.

SurgeWeb Customisation

SurgeWeb is setup for easy customisation of the basic look and feel of the interface, and basic rebranding of the interface. There is an customisation interface built in to SurgeMail that allows customisation to be done at the following levels:

- Globally - Serverwide rebranding setting defaults for all domains
- Domain - Per domain rebranding
- Group - Customisation can be applied at the surgemail g_access_group level

The surgeweb administration interface can be found under "value added features" in navigation pane of the surgemail admin interface. An autoreloading surgeweb window is available from the admin interface that will automatically refresh when any changes are saved in the customisation interface.



Reloading surgeweb interface and domain customisation disable

Customisation is applied "over the top of" the existing interface. This means it the core templates should not need to be modified and surgemail upgrades remain straight forward. If there are any interface problems it is possible to use a single domain setting to completely disable customisation on a domain for easy troubleshooting.

In the customisation interface the colour of the 'combined value' of any setting shows at what level it has been customised. For any particular setting the "deepest" level will win. So domain will override global and group will override domain.

Like old webmail Panel	Like old webmail Panel	Like old webmail Panel
global	domain	group

The documentation below shows current surgeweb operation, but everything documented here is still under development and thus subject to change.

Skinning and Interface 'Look'

Login page Appearance - A single css file allows complete customisation of the login page. 3 Examples are provided and your own css file can be specified.

Login page appearance

Like old webmail Panel

Like old webmail Panel

default

like old smooth





like old panel

'Skin' interface colouring - An interface colouring CSS can be applied to change the look of the interface. Several examples "thrown together" as follows (more will be made and can be custom made):

[Skin - interface colouring](#)

dune - color change only

dune



default + blue buttons dune loud kevin

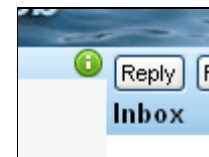
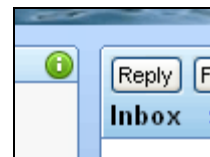
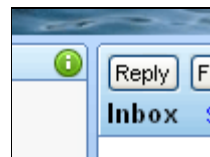
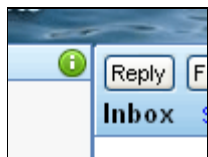
To add your own skins, add these to `surgeweb/custom/skins/` directory (surgeweb 5.0j-10+) - you can start with example and modify as needed.

Ajax 'panel look' - The panels of the Ajax interface can be setup to have gaps between them and have slightly rounded or square corners:

[Ajax 'panel look'](#)

Panelised narrow - borders between panes

Panelised narrow - Panels have borders and gaps



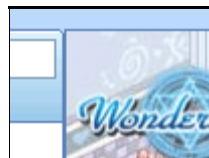

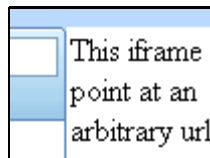
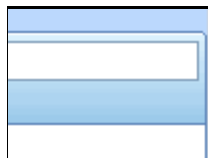
default narrow wide none

Right column content - A right column (disabled by default) can be enabled with a variety of content:

[Right column content](#)

Advertising (eg. Google text)

Text adverts, edit for your site using 'Advertising HTML' (advanced branding setting)




default custom text ads image ads

Basic Rebranding

Much of the interface can be rebranded as required using the basic rebranding settings.

- Login title** - Main login page email service title
- Login page end comment** - Comment on the bottom of the login page
- Logo image** - Logo image as displayed on the main email interface
- Info panel** - Additional panel for custom messages
- Footer text** - Text in the footer line
- Browser titlebar** - Browser page title bar

Login page customisation:		
Login title	Welcome to Xmail	Welcome to Xmail
Login page end comment	My own <a href="http://x	My own <a href="http://xyz.com" style="col...
Actual logged in email interface customisation:		
Logo image *	xmail.png upload (approx 375x50 pixels)	xmail.png 
Footer text	Xmail contact support h	Xmail contact support here: <a href="http:...
Info panel	Special <b style='color:r	Special <b style='color:red'>notification ...
Browser titlebar title	Xmail on domain \$dom	Xmail on domain \$domain\$



title



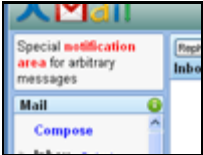
comment



logo



footer



info panel



login screen



mail screen

Note: the 'panel look' and 'background gradient image' are also customised in the above example

User Preferences

User preference defaults to be applied the first time a user logs in can be configured:

Windowing behaviour - Message editing windowing behaviour can be controlled to display and edit edit messages in their own popup windows, or in tabs in the current browser window

Windowing behaviour	Use tabbed interface	tabbed	<input type="checkbox"/>
---------------------	----------------------	--------	--------------------------



popup windows



tabbed windows

Preview window - Message preview display can be below the message list, to the right of the message list (widescreen) or completely disabled.

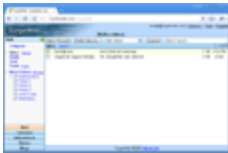
Preview window	Widescreen (beside message list)	horiz	<input type="checkbox"/>
----------------	----------------------------------	-------	--------------------------



preview normal



preview widescreen



no preview

other user preferences that can be set include:

- Auto downsize images** - Automatically downsize images when mailing large digital photos
- Date and time formats** - US / international date and time conventions
- Auto add addresses** - Automatically add messages you reply to to the addressbook & autocompletion
- Sounds notifications** - The audio feedback notifications to use for new mail and messages successfully sent
- Timezone** - User timezone
- Start 'More Folders' open** - Open or collapse the 'More Folders' grouping at login time
- Disable inbox caching** - Disable the preemptive download of the first page of inbox messages (saves bandwidth)
- Allow external images** - Conditions under which to display external images that messages link to
- Extra imap refreshes** - Do additional imap refreshes - useful if using another imap client and changes made in other client are not getting propagated to surgeweb (partially implemented)
- Initial left column size** - Initial width of the left column
- Hide integration warning** - Hide the once ever per user 'limited integration' warning regarding surgeplus / user.cgi

Other customisation

Default folder names - Surgeweb will choose the first found of several default folders for the Draft, Sent and Trash folders. The actual folder used can be set instead to a fixed folder.

Surgeweb has the facility to extend the default surgeweb behaviour using a custom css and a custom javascript file. These are advanced features for developers only and are still subject to be changed. However an example of extension is included which displays scrollable wikimapia maps in a custom pane in the left hand column:

Example extension - Enable / disable wikimapia extension - this also enables the extension css and extension javascript if they are not already enabled.

Example extension	<input type="text" value="true"/>	<input type="text" value="true"/>
-----------------------------------	-----------------------------------	-----------------------------------



Example extension

Arbitrary login screen

You can use a form on your own arbitrary web page to login to surgeweb directly. You need to pass these form fields: "username_ex", "password" and "interface_ex".
eg.

```
<html><head><title>Some arbitrary page</title></head>
<body> SURGEWEB LOGIN SCREEN
<form action="http://mydomain.com/surgeweb" method="post" name="login">
User:<input type="text" name="username_ex" value="" ><br>
Password:<input type="password" name="password" value=""><br>
<INPUT id=cmd_login name=cmd_login type=submit value="Log In">
<!-- need to pass in interface eg fixed as "ajax" or via selection
      (ajax does not work on some browsers)
<select name="interface_ex">
  <option value="ajax">ajax</option>
  <option value="html">basic</option>
  <option value="mobile">mobile</option>
```

```
</select> -->
<input type=hidden name="interface_ex" value="ajax">
<!-- Optional pass in the domain to login as - normally identified from url
<input type=hidden name="domain_ex" value="mydomain.com" -->

</form>
</body></html>
```

The above will work regardless of url, provided the form gets directly submitted to surgeweb from the browser. If instead you wish to setup surgeweb as part of a central login service (generally on a url separate to the surgeweb host name) this is documented on the [clustering page](#).

Login page links

The following account management links can be added to the surgeweb login page, by manually adding these settings to config_*.dat files. (surgeemail 5.0j-9+)

```
# Forgot password link
showlink_forget_pass true
# New account creation link
showlink_create_account true
# Custom help link
showlink_login_help true
login_help_link http://myserver.com/myhelp/pages.htm
```

SurgeWeb Contacts

There are three aspects to the surgeweb contacts interface:



As you use surgeweb the addresses of the people you email will automatically get added to the contacts list and you will probably find you will just use the autocompletion. The picker can be used to add recipients or groups to an email when composing a message and the management page is used to edit and organise your contact information.

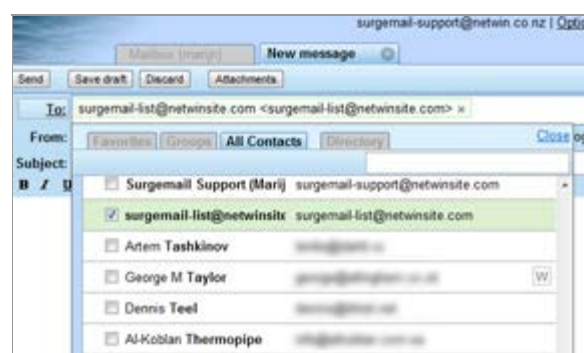
Autocompletion and recipients field



Not much to say here, it's autocompletion... you start typing a few characters, surgeweb autocompletes and you pick the address by mouse or by keyboard.

Already added recipients can be removed by pressing the small cross.

Contact picker



The picker allows you to quickly select and add multiple recipients to your email. Press the To button, optionally filter by search criteria, and select the addresses you want to add.

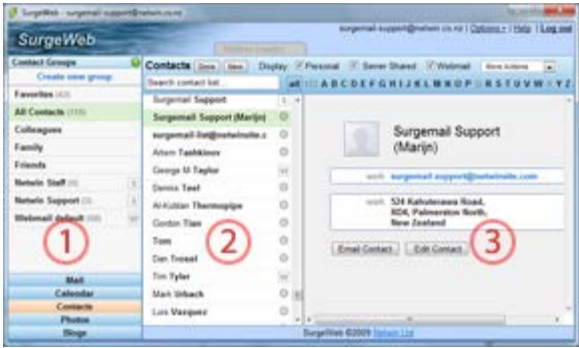
Three different sets of contacts are currently selectable:

- Favorites - this is the list of 20 most emailed contacts ever, plus the the 20 most recently emailed contacts, plus any you have manually added.
- Groups - Groups can be added to the the recipient list. It is possible to see who is in each group before adding the group.
- All contacts - All the contacts - dependant on the contacts source on the management page this will be the

email addresses in your personal contacts, plus all shared addressbooks, plus any webmail addressbooks you may still have.

The contacts management page

The contacts management page allows you to do various more detailed contacts management functions. This includes organising contacts into groups (aka distribution lists), manually add or remove contact information, edit contacts providing additional information.

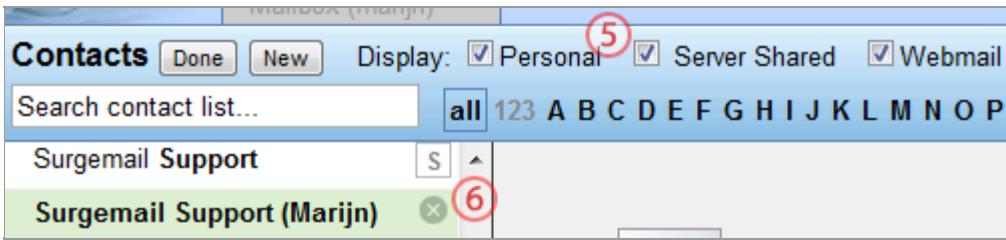


The page contains 3 main panes:


- 1. Left hand pane listing various forms of groups of contacts
- 2. Center pane of all the contacts in the group selected in the left
- 3. Right hand pane of detailed contact information for a particular contacts.

Source of contact information

Surgeweb conceptually works with a single list of contacts that gets sourced from one of several locations. Primarily your surgeweb personal contacts (see 5 below). However in addition there may be access to shared contacts, and any webmail contacts. These sources can be individually enabled using the appropriate tickboxes.

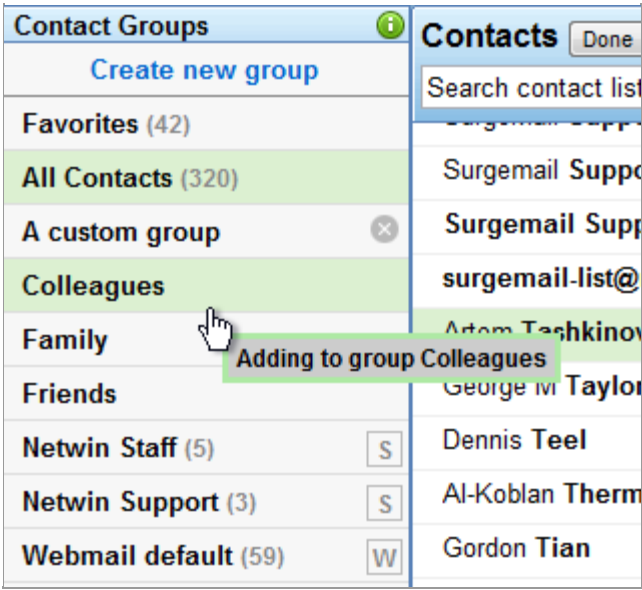


The source of individual contacts is displayed (see 6 above) as part of the list as per icon. Webmail contacts are always readonly in surgeweb. Shared contacts are readonly unless abook rules have been defined that specifically give you permission to edit the shared addressbook.

 personal surgeweb contact (editable)  Server shared contact  webmail contact

Groups

A group is a grouping of contacts for easy management. A group can be used as a recipient so can be treated as a distribution list. To organise contacts into just drag and drop a surgeweb contact (ie with a round circle) into the the group of choice.



Some other important aspects of groups:

- The same contact information can be part of multiple groups
- The "Favorites" special group automatically contains most frequently and recently mailed addresses. Other addresses can be manually added.
- The "All Contacts" is a list of all the contacts surgeweb knows about.
- The default groups "Colleagues", "Family" and "Friends" cannot be deleted.
- Any Webmail addressbooks (loaded as read only) are listed as a group
- Groups may also be defined in shared addressbooks

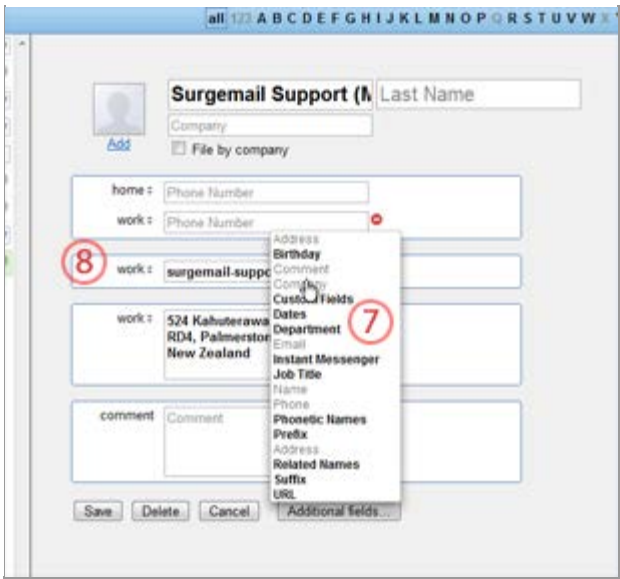
Searching / filtering the contacts

The center pane will list all the contacts in the currently selected group. These can be filtered using the Search field or the index tabs.



Editing contact information

A good selection (see 7) of additional fields is available for adding detailed contact information. For most fields multiple fields can be defined and the "type" of each field is selectable (see 8) eg home, work, etc.



Importing webmail contacts

Webmail contacts are normally displayed as readonly entries in the surgeweb contacts. When a webmail contact is displayed in the details pane, links should be displayed (see 9) that allow just this contact or all webmail contacts to be imported into surgeweb. If all webmail contacts are imported the display of webmail contacts is disabled to avoid entries which are effectively duplication of the same information.



Importing CSV contacts

Outlook formatted contacts CSV files can be imported using the Import option in the "more Actions" menu.

Unimplemented features in the menu

Some features in the "more Actions" menu remain unimplemented for now. These will be implemented as soon as we can manage :-)

Shared contacts (administrator information)

In addition to users own personal contacts, multiple shared addressbooks are available. These can be edited using the surgeweb contacts interface providing permission has been enabled to do so.

Permissions to view and edit shared addressbooks is defined using the abook domain settings in surgemail.ini. These abook entries can also be used to define additional shared addressbooks. By default surgeweb will try and serve a global and domain addressbook as part of a users contacts without the need for abook settings. These are the same format as the user's user.abk file and stored in:

```
surgemail/abook/Global.abk
surgemail/abook/mydomain.com/Domain.abk
```

You do however, need to add abook settings for the Global and Domain addressbooks in order to edit these using the surgeweb interface.

Here are some possible example abook settings:


```
abook name="Global" read="*" write="surgeweb_admin"
abook name="Domain" read="*" write="surgeweb_admin"
abook name="Special" read="specials" write="special_admin"
abook name="WideOpen" read="*" write="*"

```

The above settings allow:

- All accounts to see the global and domain addressbooks. And accounts with "surgeweb_admin" access groups to edit the global and domain addressbooks.
- In addition, a custom addressbook is defined named Special. Any accounts with "specials" groups would get these contacts displayed, This is only editable if an account has the "special_admin" permissions.
- Lastly a custom addressbook is defined named "WideOpen" which any user can edit and will get displayed in every users contacts. (dangerous!)

Note: The above read and write group names are arbitrary and must be defined using g_access_groups.

Users' surgeweb personal contacts are stored in the user.abk file stored in the users' mailstore mdir folder. These are stored in a similar but extended version of the nwauth database files. In fact you could use an nwauth.txt or nwauth.add file directly as a starting point for a surgeweb addressbook - this is particularly useful for getting a shared addressbook started.

Alternatively use the "New contact in..." and "Copy contact..." under more actions menu to get started editing shared addressbooks.

Further additions of being able to talk to an LDAP addressbook, and possibly the current authentication database are planned in the near(ish) future.

SurgeWeb User Help

SurgeWeb end user help.

Labels

SurgeWeb has two slightly different forms of labels with comparative advantages as below:

Universal labels

Should be used for labels you use frequently and may want to change on multiple messages. This uses IMAP user flags to store the labels.

- This is efficient
- Labels can be viewed with another IMAP mail client
- There is a limit of 22 of these labels - which includes any your IMAP mail clients may create
- Custom IMAP flags must be explicitly enabled by the server administrator using g_imap_user_flags

SurgeWeb labels

Only use if you need more than 22 labels or imap user flags cannot be enabled. Avoid using these on large messages. This stores the label as an additional header in your IMAP message. This requires surgeWeb to modify and re-upload your email message to the server.

- There is no limit on the number of labels that can be created
- This is less efficient (particularly on large emails), and can only be set on a limited number of messages at a time
- Labels can only be viewed from surgeWeb
- If you do a "forward attach" the labels will be sent as part of the message headers so the recipient may be able to view them (just like labelling using Thunderbird in POP mode)

Labels can be added and removed from messages as much as you like. However for both types of labels once the label has been created it cannot be deleted (for implementation reasons). You can however rename the display name or hide labels from display.

Both labelling mechanisms should work across surgemail mirrored servers.

Searching

SurgeWeb has three different ways of searching for messages

Quick search

This is a 'search as you type' browser side 'full text match only' search of the displayed headers of the current page of messages. This is useful for quickly locating certain messages without waiting for the delay involved in going to the server for more advanced searches.

Note: This will not search message bodies or other pages of messages in the currently displayed folder (see hint below)

Full text match only:	
joe	All messages with the word 'joe' in any of the cached headers
joe blogs	All messages with the string 'joe blogs' in any of the cached headers

HINT: When Enter is pressed a "quick search" will be automatically switched to a "folder search" under certain conditions. In particular: if there are multiple pages of messages in a folder, if advanced search syntax characters are found (colon or minus or double quote), or control-enter is pressed.

Folder search

This is a very fast server side headers search of all the messages in one or more folders. This search capability allows for the search for multiple search terms that are 'ANDed' together and allows for searching in in specific fields:

Note: This will not search message bodies, and only searches messages in folders that have already been accessed and indexed by surgeweb (manually refresh by right clicking a folder and select refresh or refresh all)

Basic text searches:	
joe	All messages with the word 'joe' in any of the cached headers
joe blogs	All messages with the word 'joe' AND 'blogs' in any of the cached headers
joe blogs -foobar	All messages with the word 'joe' AND 'blogs' EXCLUDING 'foobar' in any of the cached headers
"joe blogs"	All messages with the string 'joe blogs' in any of the cached headers
from:"joe blogs"	All messages with the string 'joe blogs' in the from header

Specific field searches:	
email:joe@domain	All messages with 'joe@domain' in any of the recipient fields ie. a "conversation history"
from:marijn@netwin	All messages with 'marijn@netwin' in from address
to:joe	All messages with 'joe' in to address
subject:webmail	All messages with 'webmail' in the subject ie. a "thread view"
date:week	All messages received in the last week valid values: day, week, month
flags:replied	All messages that have been replied to valid values: seen, replied
attachments:any	All messages with attachments valid values: any, true, none, false

More complex searches:	
from:marijn@netwin date:week	
	All messages from 'marijn@netwin' in the last week
subject:webmail -date:week	
	Messages with the word 'webmail' in subject older than one week
from:joe subject:webmail	
	Messages from 'joe' with the word 'webmail' in the subject
-flags:replied -date:week	
	Messages older than one week that have not been replied to

note: The "*Recent*" option only searches folders that have been accessed in the last month. The "*All*" option will search all folders but does not refresh the message indexes from IMAP so may not find the message you are looking for. Click any folders in question to refresh the indexes.

Body search

IMAP serverside search of the full message body and headers content. This can take a long time on large folders /

or large accounts - expect approx 30 seconds per 100 MB of mail that needs to be searched through. One or more search terms can be specified which will be ANDed together.

Note: This may take a long time

Body search syntax:

- joe All messages with the word 'joe' anywhere in the message
- joe blogs All messages with the word 'joe' AND 'blogs' anywhere in the message
- "joe blogs" All messages with the string 'joe blogs' anywhere in the message

Spam handling

SurgeWeb uses the surgemail user.cgi Friends and spam handling features. Manually correcting false positives and training messages as spam can be done using the following actions in SurgeWeb or in an IMAP client:

	Action	*** subject marking	Move to	notspam train	isspam train	Contacts add	Friends whitelist	Address blocklist
SurgeWeb - Buttons on messages having received high spam rating: (single message only)								
	Press "Spam" button		Spam		Train		Remove	
	Press "Not Spam" button	Cleaned		Train		Add	Add	
	Press "Allow Once" button	Cleaned						
	Press "Block Forever" button		Spam				Remove	Add
	Press "Spam" button		deleted		Train		Remove	
	Press "Not Spam" button	Cleaned	INBOX	Train		Add	Add	
	Press "Allow Once" button	Cleaned	INBOX					
	Press "Block Forever" button		deleted				Remove	Add
SurgeWeb - "More Actions" menu action: (one or more messages)								
	"Spam" whilst in any folder other than Spam		Spam		Train		Remove	
	"Not Spam" whilst in any folder other than Spam	Cleaned		Train		Add	Add	
	"Spam" whilst in Spam folder		deleted		Train		Remove	
	"Not Spam" whilst in Spam folder	Cleaned	INBOX	Train		Add	Add	
SurgeWeb - Drag and drop / copy / move: (one or more messages)								
	from any folder to Spam folder	no special spam handling actions taken						
	from Spam folder to INBOX	Cleaned	INBOX	Train			Add	
	from Spam folder to other folder	no special spam handling actions taken						
IMAP copy / move: (one or more messages)								
	from any folder to Spam folder	no special spam handling actions taken						
	from Spam folder to INBOX	Cleaned	INBOX	Train			Add	
	from Spam folder to other folder	no special spam handling actions taken						

Primary action ☐ actioned in INBOX or other folder ☐ actioned in Spam folder

Multilanguage support

Surgeweb now has extensive multilanguage support throughout the interface (as of surgemail version 4.2g-22+) to customise the interface for both for English and non English languages. Most of the strings displayed in the surgeweb interface can now be translated or customised (with a few exceptions for now, as noted later on).

Surgeweb language translation features are similar to the language translation capabilities of the other surgemail web interfaces, but are notably different in a few areas.

- Implementation is string table based, which puts all languages - English and others - on an equal footing. This means actual english phrases can be modified without it "breaking" the translations for all other languages. Also it makes it easy to find the strings to translate.
- Customisation is hierarchical, and allows changes without modifying the base implementation. This is much like all the customisation concepts in the rest of surgeweb.
- Distinct support for UK / US English.
- Languages selection is displayed in native text rather than English. eg. Français instead of French
- There are sample "partial translations" (some shown below) of 26 non English languages to allow you to get an idea what the surgeweb interface is capable of and would look like with a full translation. A full translation requires approx 800 phrases to be translated, although you can just translate a subset if you wish. The sample translations have only 16 phrases translated.



How multilanguage support works

The surgeweb template files contain placeholder strings along the lines of `$$st_cmd_reply$$` which are string table lookup entries that get replaced with the actual textual representation before getting served to the web browser. So in the case of the Reply button, this would be replaced with "Reply" in English or with the string "Antworten" if German was selected.

Reply button as defined in the template files:

```
<input type=button value="$$st_cmd_reply$$">
```

Sample lang*.dat file format:

```
# notel: Any line starting with a # is a comment
# note2: lang*.dat files need to be UTF8 encoded

# Language definition of the way languages get displayed in the interface
# - both the english name and the native display name
# - type of translation:      *=full translation,
#                             o/+ =partial sample translation (under dropdown/more...)
#                             --hide interface
# Here English is a full translation and german and spanish demo translations
lang*:English:English (UK)
lang:o:German:Deutsch
lang:+:Spanish:Español

# -----
lang:English
# -----

# Strings to add to English, individual strings can be added to alternate languages
st_username      Username
st_password      Password
```

```
st_cmd_reply      Reply
    German:      Antworten
st_cmd_forward    Forward

# Generally alternate languages are listed as complete standalone tables,
# possibly in their own files. "lang:" must match a full lang line at top of file.
# -----
lang:Spanish
# -----
st_username       Nombre de usuario
st_password       Contraseña
st_cmd_reply      Responder
st_cmd_forward    Reenviar
```

These surgeweb string tables are defined using the **sum total of lang*.dat files** in the **surgemail/surgeweb/tpl** and the **surgemail/surgeweb/custom** directories. Language tables in the tpl directory is loaded first, and this is overwritten by any customisation as defined in the custom directory. This allows you to implement full translations, or modify any terms or full translations as supplied with surgeweb. Anything in the tpl directory will be replaced on surgeweb upgrades, none of the language files in the custom directory will be touched during surgemail upgrades.

List of lang files supplied with surgeweb

- **tpl/lang.dat** - Core surgeweb interface files defining "English UK" and "English US". Essential for surgeweb operation.
- **tpl/lang_demo.dat** - Demo partial translations of 16 phrases in 26 languages. Optional, can can be disabled as noted below.
- **tpl/lang_east.dat** - Empty placeholder for full translations of the "eastern" languages.
- **tpl/lang_euro.dat** - Empty placeholder for full translations of the "european" languages.
- **tpl/lang_asia.dat** - Empty placeholder for full translations of the "asian" languages.

Recommended language customisation files

- **custom/lang.dat** - Recommended primary file to customise / add translations (probably all you need for language translation).
- **custom/lang_mylang1.dat** ... - If you are doing full translations for several languages it is nice to keep these as fully specified tables in separate files per language.

Some other important things to know:

- In order to apply any changes in the language translation tables you need to restart surgemail.
- The default language that surgeweb / user.cgi / surgeplus defaults to can be set using the English language name in the (g_)language_default surgemail.ini setting.
- Dependant on how the strings are used in the templates, single quotes will generally need to be encoded as "\" to prevent javascript errors.
- Some string parts get replaced with variables such as username, links etc. These are marked as "%s" in the strings. These can be safely omitted from the translations, if you do not wish to provide these as part of your custom strings.

If you are interested in supplying us with a translation of a non english languages we would be very keen to hear from you :-)

Customising / translating SurgeWeb

I just want to customise the English phrase "x":

Search through tpl/lang.dat to find the phrase(es) in question.

eg. Change "Loading SurgeWeb email ... step n of 3" whilst logging in to "Please wait :-)"

```
st_loading_init    Loading SurgeWeb email ...
st_loading_usr     Loading SurgeWeb email for %s@%s ...
st_loading_step    step %s of %s
```

and add that to an appropriate custom/lang.dat file: (note: add to blank out phrases)

```
# -----
lang:English
# -----
st_loading_init    Please wait :-)
st_loading_usr     Please wait :-)
st_loading_step    &nbsp;
```

I want hide all demo languages

Using the admin interface enable / disable "Hide Language Demos" globally / per domain / per group. This adds the following setting to custom/config*.dat.

```
lang_no_demo true
```

I want to turn a language "x" into a more complete translation

Open tpl/lang.dat and copy just the strings you want to translate or the whole table and copy to custom/lang.dat or say custom/lang_german.dat. Translate all phrases you wish nationalised. Any phrases not translated will get displayed in surgeweb as default English UK. Also at the top add a lang line with 'o'/'+' converted to '*'. This means surgeweb will treat this language a full translation rather than a partial demo translations.

eg. in the case German:

```
lang:*:German:Deutsch
# -----
lang:German
# -----
st_username      Nutzername
st_password      Passwort
st_log_in        Anmelden
... etc
```

I want hide several demo languages but keep others

Copy the "lang lines" from tpl/lang.dat to custom/lang.dat and change the 'o' or '+' to '-'. eg. to remove Chinese and Japanese:

```
lang:-:Chinese 1:中文 繁體
lang:-:Chinese 2:中文 简体
lang:-:Japanese:日本語
```

I want to change the default language to Xxx

To change the default language you can use the English language text description (as per lang*.dat files) in the surgemail.ini (g_)language_default settings.

eg. to switch to US English default:

```
g_language_default "English US"
```

or Japanese

```
g_language_default "Japanese"
```

Note: The language selection default is cookie based, and will not apply to users that have already connected to surgeweb unless they clear the cookies in their browser cache.

Still to be completed...

So far most of the phrases (approx 800) that have always been defined in the surgeweb templates are now customisable this way.

As noted earlier, for now there are several exceptions, which I will be adding to the language translation capabilities in the near future. These include:

- Surgemail serverside generated error codes
- A few items in the folder and contacts tree views (notably folder names and group names)
- User.cgi / surgeplus generated pages & information. These pages support the translation mechanisms as they have always done, but no work has been completed yet to fully integrate these features with surgeweb. Although the basis for the translation - a cookie named webmail_lang - is fully two way compatible between these translation mechanisms.

Customer contributed translations

Surgeweb [language translation files](#), as contributed by customers.

Recommended "Surgeweb Style" spam handling

Surgeweb attempts to simplify surgemail's spam handling for end users and administrators. This involves storing all suspected spam in a single location, and having several predefined recommended configurations for users to select.

All of the traditional user.cgi spam handling feautes are still available, but most users should never need to configure them directly.

Recommended configuration in Surgeweb

There is a "one step spam control" that allows you to configure all the underlying user.cgi spam handling settings in one of several configurations depending on your personal preference on how you want email identified as spam to be handled:

Options

Save

Cancel

General

Filtering & Spam Control

Screen Layout

Customise

Advanced

DEVELOPER

One step spam configuration:

Challenge almost certain spam, mark likely spam RECOMMENDED

If you receive lots of spam, the recommended spam handling is to *split probable spam into*:

- *likely spam* - which should be checked occasionally for false positives - subject marked in your inbox

- *almost certainly spam* - which you probably need never look at - delivered to your spam folder with friends challenge for delayed delivery to inbox

Check the [delivery log](#) for a record of processing actions.

Received mail gets processed in the following order before delivery to the inbox.

<div><div>Holiday autoresponder</div><div>Automatically send a reply to received messages</div></div>	<div>INACTIVE</div> <div>configure</div>
<div><div>Forwarding</div><div>Send message to on other email accounts</div></div>	<div>INACTIVE</div> <div>configure</div>
<div><div>Filtering rules & exceptions</div><div>Filters to organise mail and bypass spam & unknown sender controls</div></div>	<div>INACTIVE</div> <div>configure</div>
<div><div>Safe address whitelist (Friends list)</div><div>Safe sender addresses to bypass spam & unknown sender controls</div></div>	<div>ENABLED</div> <div>configure</div>
<div><div>Spam control</div><div>Identify and process suspected spam messages</div></div>	<div>ENABLED</div> <div>configure</div>
<div><div>Challenge unknown senders (Friends bounce)</div><div>Action to take if sender of message is unknown</div></div>	<div>ENABLED</div> <div>configure</div>

Raw user.cgi spam related settings (directly editable via advanced settings):

CORE SETTINGS: spam_subject=4 friend_mode=smite friend_smite=10 spam_bounce=0

ADDITIONAL: spam_body=0 spam_ddpriv= spam_ddfrom= [spam_store=0 spam_vanish=0]

These are the recommended configuration "one step spam control" configurations available:

1. **Disabled** - Surgemail does not identify or process mail in any way as spam. Not recommended.
2. **Mark probable spam and deliver to inbox** - As the name suggests, email with a rating of 6 or higher is marked in the subject and delivered as normal to the inbox. Suitable if you receive small amounts of spam.
3. **Mark probable spam and deliver to spam folder** - Again, email with a spam rating over 6 is marked in the subject but delivered to your surgeweb Spam (ie IMAP Friends Pending) folder. Suitable if you receive small amounts of spam, want to keep this "out of your inbox" and are happy to periodically check your Spam folder.
4. **Challenge almost certain spam, mark likely spam** - A more advanced way of spam handling if you receive lots of spam. Email with a spam rating is split into "likely spam" (rating 6-10) and "almost certainly" (rating 10+) spam. Likely spam is marked in the subject and delivered to your inbox as normal, and almost certainly spam is sent a friends bounce and placed in your Spam folder. You will probably never need to look in your spam folder, and if real mail is caught and the sender replies to the friends bounce the message gets delivered to your inbox. Under the "surgeweb - options - Filtering and spam control - spam control - configure" link there are more options for configuring your advanced "split spam handling" configuration.

Administrators can configure the cutoff levels for "likely" and "almost certainly" spam. If you can't figure out what is going on the raw user.cgi settings are listed at the bottom of the "Surgeweb - Options - Filtering & Spam

control" page. For each configuration these are:

1. Disabled

```
CORE SETTINGS: spam_subject=0    friend_mode=list    friend_smite=0    spam_bounce=0
ADDITIONAL:   spam_body=0        spam_ddpriv=      spam_ddfrom=      [ spam_store=0    spam_vanish=0 ]
```

2. Mark probable spam and deliver to inbox

```
CORE SETTINGS: spam_subject=6    friend_mode=list    friend_smite=0    spam_bounce=0
ADDITIONAL:   spam_body=0        spam_ddpriv=      spam_ddfrom=      [ spam_store=0    spam_vanish=0 ]
```

3. Mark probable spam and deliver to spam folder

```
CORE SETTINGS: spam_subject=6    friend_mode=silent    friend_smite=6    spam_bounce=0
ADDITIONAL:   spam_body=0        spam_ddpriv=      spam_ddfrom=      [ spam_store=0    spam_vanish=0 ]
```

4. Challenge almost certain spam, mark likely spam

```
CORE SETTINGS: spam_subject=6    friend_mode=smite    friend_smite=10    spam_bounce=0
ADDITIONAL:   spam_body=0        spam_ddpriv=      spam_ddfrom=      [ spam_store=0    spam_vanish=0 ]
```

Upgrading from "older" surgemail configurations

If are upgrading an older surgemail configuration to make use of surgeweb new features you may want to make some configuration changes. In particular the recommended method of [quarantening spam](#) has changed a little over time.

You will probably want to [convert existing accounts](#) using the "tellmail held2pend ..." command, and configure the new global default behaviour:

```
g_friends_default_mode "silent"
```

Plus optionally:

1. Set default friends spam scoring to match surgeweb ratings likely="6+" almost certainly="10+" This can be done using a global setting as follows in surgemail 4.2g-2+,

```
g_friends_spam_score "6"
```

or

```
g_friends_spam_score "10"
```

or in earlier versions of surgemail, using the global user defaults on the accounts page in the admin interface.

2. To make the "(Friends) Pending" folder available as the "Spam folder" in surgeweb (HIGHLY RECOMMENDED), make sure you have enabled this using:

```
g_imap_friends "TRUE"
```

3. If you wish the Imap folder name (normally "Friends Pending") to match the surgeweb folder name (normally "Spam") use:
surgemail.ini:

```
g_friends_pending_name "Spam"
```

surgemail/surgeweb/custom/config_global.dat:

```
imap_spam_folder Spam
```

In addition you can customise the "levels" that define "likely spam" and "almost certainly" spam by customising these manually in the surgeweb config_*.dat files. It is recommended to select a levels the same for all domains and make sure this matches g_friends_spam_score above for the spam settings for new accoutns to fall into one of the recommended "one step" surgeweb spam handling configurations.

```
rating_probable 6
rating_certain 10
```

New format html spam status email

The old plain text status email reporting on the status of quarantened spam has been replaced by an html formatted email with more information on the email messages that have been detained in the Spam folder (ie Friends Pending folder). This html status email provides the following features:

- 'Single click deliver' messages to your inbox, training the from address in your friends list to ensure future messages from this user are always delivered without being marked as spam
- 'Single click view' the message safely without delivery to your inbox
- 'Single click delete' the message
- Permanently block all email from this sender
- There are also options to enable spam reporting and Spam folder purging from this page
- Displays all the relevant addressing headers if they are different from the From header to help identify mail with faked headers. In particular "Reply-To" and "Return-Path".

From: Mail Delivery Subsystem <postmaster@netwin.co.nz>
Subject: Spam Report for support-pn@netwin.co.nz (29 messages)
Date: Saturday, 24/10/2009 2:09 PM

Show: rawhtml

This message is an automatic status message summarizing messages detained as suspected spam and a log of important serverside email processing events. To disable or configure this message login to the [user.cgi interface](#) and click on the "Log" link.

29 new suspected spam (Total messages: 499)		1 day(s) to Friday, 23-Oct-2009 20:09:06 CST	
Action links	Score	Message details	Size Expiry
Deliver to INBOX View Delete Block	13	Subject: Greetings From London & Business Proposal From: workjim@virgilio.it (Jim McConville) To: workjim@virgilio.it Reply-To: tkbnjim@yahoo.co.jp	1 KB 14 days
Deliver to INBOX View Delete Block	13	Subject: Greetings From London & Business Proposal From: workjim@virgilio.it (Jim McConville) To: workjim@virgilio.it Reply-To: tkbnjim@yahoo.co.jp	1 KB 14 days
Deliver to INBOX View Delete Block	30	Subject: Enjoy a new car, no matter your credit From: wyome.vto1638@freeautumndirect.com (AutoLoans) To: <support-webmail@netwinsite.com> Return-Path: bobxf@freeautumndirect.com	1 KB 14 days
Deliver to INBOX View Delete Block	20	Subject: Get a diploma for a better job. From: dichotomiesld@800democrats.com (Joni Salinas) To: <support-webmail@netwinsite.com>	1 KB 14 days

For more information see [recommended method](#) of quarantening spam.

SurgeWeb Clustering

There are several different ways to configure surgeweb depending on what you are trying to achieve.

The core of surgeweb is conceptually really just a "serverside normal IMAP client" and should be able to talk to any IMAP server. In addition, some features require surgeweb to be talking to a surgemail backend server - these include the contacts features, surgeplus, and user.cgi features such as spam handling.

Single Server

This is the default normal surgemail and surgeweb deployment. Both surgeweb and IMAP are hosted by one surgemail process. Surgeweb will talk to imap on the same server as necessary.

No special configuration is necessary other than a normal surgemail install.

Frontend / Backend Configuration

In this configuration, surgeweb talks directly to the backend imap server. The server running Surgeweb only hosts temporary files and all the mail, surgeweb configuration files, and the surgeweb contacts (personal and shared) are stored on the primary IMAP server. There are two main reasons that you may want to split the default configuration into a frontend / Backend configuration:

Spreading of load

By installing surgeweb on a separate server all surgeweb related load will be handled by one or more frontend servers, and only a minimal loading is placed on the main server. Surgeweb is designed to minimise the load on the IMAP server - using surgeweb as a client actually provides a lot lighter loading on the imap server than a normal desktop IMAP client.

Reliability & Testing

Under single server configuration, any problems in surgeweb that result in a surgemail restart will affect all mail services. By running surgeweb on a separate server most problems you might encounter in surgeweb will only affect any other surgeweb users actually using the system.

Also, as surgeweb is still under active development and a lot of changes are still being made you are likely to want to upgrade surgeweb more frequently than your primary mail server. The preferred way is to use mirrored systems and installing the latest version on the mirror, have your "test users" using the mirror and if all proves well upgrade the primary system. Alternatively you can run surgeweb on a frontend system and have this pointing at your primary server.

To configure a frontend system:

1. Make sure your backend system has surgemail fully configured and operational.
2. Install surgemail on the frontend server and make sure that it is configured pretty much identically to the backend system. Primarily you just need the same vdomains defined, but it is recommended to have the global and per domain settings as similar as possible to the primary system.
3. On the frontend system you just need to add the domain settings:

```
surgeweb_backend_server "ip.of.main.server"
```

and if non default you may need settings along these lines:

```
surgeweb_backend_smtp "ip.of.smtp.server"
surgeweb_backend_web "ip.of.main.server"
surgeweb_backend_web "https://ip.of.main.server:port"
```

4. Now test that everything is correctly configured by logging on to surgeweb on the front end system and seeing if you see the correct details of the mail account on the backend system. Also test the contacts interface and usercgi/surgeplus functionality to make sure it is autologging into surgemail correctly.

Note: This can be used to have surgeweb connect to arbitrary imap servers, however if the backend servers are not surgemail dealing with email should generally work, but surgemail only functionality will not work. (In most

cases it will be visible in the interface but just not functional).

eg

```
surgeweb_backend_server "imap.gmail.com:993"  
surgeweb_backend_smtp "smtp.gmail.com:465"
```

SurgeWall Configuration

In this configuration surgeweb is again part of surgemail installed on a separate server. The surgemail server logically sits "in front of" the backend mailserver that is actually storing the mail. In this way the surgewall server can provide mail filtering services and provide surgeweb access to the mail stored on the primary server.

All the surgeweb configuration files and contacts etc are stored on the surgemail server. Surgeweb connects to imap on the surgemail that is hosting surgeweb, any normal imap commands are proxied on to the backend server and any surgemail functionality - like contacts handling - is handled by surgemail.

For more information on surgewall see [surgewall documentation](#)

This is not yet fully functional and will be implemented in the near future. At the moment this will behave in much the same way as a frontend system talking to a non surgemail server.

Please contact us on surgemail-support@netwinsite.com to up the priority on this if you are actively waiting for this.

Using a Central Login with Surgeweb

For a simple login to surgeweb from an arbitrary web page this can be done as noted on the [customisation page](#). If you wish to setup a central login that will allow you to maintain logged in state and to allow you to login to several websites including surgeweb, a different approach is required.

Direct links:

The first (STRONGLY DISCOURAGED) option is to pass out links or form posts that will submit a form with the username and password. This requires both the username and password to be sent to the browser, is very insecure, and may even result in the users credential getting displayed in the browser bar or referrer logs.

Redirection

A second option is for your central login cgi to maintain logged in status / ask for username password etc, and if it is determined that you are currently "logged in" to the central login service, to connect to surgeweb with the username and password and serve the contents of the following web request:

eg browser connects to

```
http://login.mydomain.com
```

which runs your very own login cgi that maintains central login state. If it determines you are logged in, it connects to surgeweb using this url (unwrapped), and serves the data that it receives back from surgeweb directly.

```
http://mail.mydomain.com/surgeweb?username_ex=user@mydomain.com&password=mypass  
&cmd=login,show&page=external_login.htm&interface_ex=ajax
```

This will result in a redirection at the surgeweb level to correctly set the authentication cookie, and will result in a logged in surgeweb session. The actual user's password is only passed directly from your central login service to your surgeweb server, and is never used browserside where it may be a security risk.

note: Old style webmail autologin notes can be found in the [webmail help](#). I may be implementing similar functionality for surgeweb although the above results in much the same overall effect.

note: Also note that surgeweb actually has built in cookie based "remember me" functionality that will keep you logged in for several weeks.

SurgeWeb Performance

The SurgeWeb performance relative to webmail is very good. Here are some example statistics that are a month or two old but should still be mostly relevant. I'll update them at some stage soon.

Server side measurement

Individual serverside request duration on a real account with a moderate number of folders, all associated settings files, addressbook files etc. Tested on a server with a trivial disk and network io loading.

Action	SurgeWeb	Existing Webmail
Page refresh of inbox with 10 messages	390 ms with imap refresh 20 ms from surgeweb cache without imap refresh	1300ms
Recheck mail with empty inbox	300 ms with imap refresh	1200ms
Very simple text message display	210 ms (with IMAP get) 13 ms (from surgeweb cache)	2600ms (with IMAP get) 1600ms (from webmail cache)
Get 10kb attachment part for already displayed message	7 ms	200ms
Folder with 10500 real messages	first click: 3740ms (IMAP download of 200 msg) background folder update (~60s for all 10500) once all messages in index, next page: 400ms (or 1200ms with imap check of 100 msg)	first click: 7500ms (IMAP download of 200msg) many page refreshes later all are downloaded once all messages in index, next page: 6900ms

Client side measurement

Client side bandwidth and response time measurements.

Note: Network characteristics: ADSL network connection with me in NZ and server in US. Server ping time 230ms. Downloads achievable of 200KB/s so no apparent network bottleneck getting hit.

First login large account (nothing in browser chache):

```
SurgeWeb - already has inbox from imap, includes all js/css/audio for
whole interface and twice as many messages as the webmail equivalent
Ajax: 6.0 seconds; 12 requests; 49KB received
      (image=9kb, css=3kb, html=4kb, js=30kb) 8kb sent
      + 1 request for inbox caching if enabled - for typical 100 messages
        in surgemail support 60kb (expands to 350kb uncompressed)

Webmail - already logged in before, already has inbox from imap
          (300msg) 50 msg displayed on page
Surge: 6.5 seconds; 7 requests; 70KB received
      (image=3kb, css=5kb, html=50kb, js=12kb) 5kb sent
Smooth: 21 seconds; 60 requests; 370 KB received
      (image=10kb, css=15kb, html=223kb js=106kb) 40kb sent
Panel: 12 seconds; 47 requests; 376 KB received
      (image=40kb, html=293kb js=29kb) 38kb sent
```

Second login, same account (most things in permanent browser cache):

```
SurgeWeb - already logged in before, already has inbox from imap
          (300msg) 100 msg displayed on page
Ajax: 2.5 seconds; 2 requests; 9kb received; 1kb sent

Webmail - already logged in before, already has inbox from imap
          (300msg) 50 msg displayed on page
Surge: 2.2 seconds; 1 request; 49 KB received; 1kb sent
Smooth: 10 seconds; 3 requests; 223 KB received; 3kb sent
```

Panel: 6.2 seconds; 4 requests; 294 KB received, 4kb sent

Note 1: These were once off measurements to get an indication of surgeweb performance

Note 2: Since these measurements were taken the most significant thing that has changed is that the initial javascript download of the surgeweb Ajax interface has increased from 30kb to around 50kb.

Written late 2008, further measurements will be made in the near future as needed and as time permits.

[Index](#)

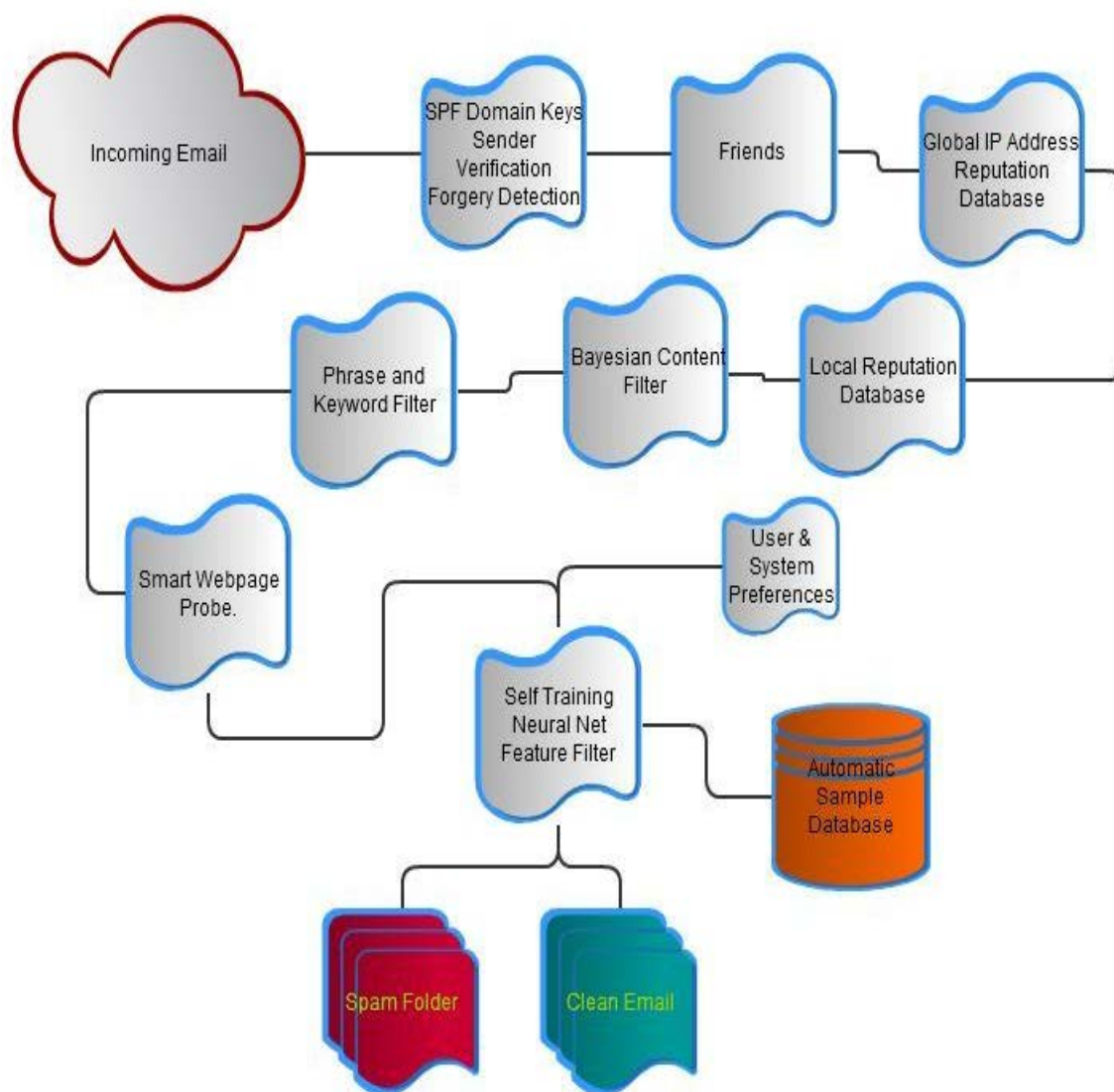
Major topics.

- [How it works](#)
- [How to turn it on](#)
- [How to tune it for your users \(help I've got more spam, or more bounces\)](#)

Technical Info:

- [New commands and settings](#)
- [Brief outline of how a message is processed](#)
- [Features to stop cracking local accounts and sending out spam](#)
- [Explanation of the X-SpamDetect header](#)
- [How to disable the new system!](#)
- [How to convert your local.rul file to sf_mfilter_local.txt](#)

MyRBL / Smart Filter -- New Spam Handling Process



How it works - Summary

Each incoming message is analyzed for as many 'features' as can be found that have significance, then the pre generated feature processor rule file combines those features to figure out the probability that the message is SPAM or not. If the message passes a minimum level then it is placed in the users 'spam' folder and the sender is notified by a url (with CAPTCHA) this allows a real sender to bypass the 'spam' folder and ensure delivery.

The user can also access their spam folder via IMAP or the SurgeWeb email interface if they suspect a message may have been miss classified

This system has several key benefits:

1. No messages are blocked or dropped, failed deliveries due to filtering mistakes are enormously annoying, and no filter is ever accurate enough to allow this type of heavy handed mechanism.
2. The user does not 'see' the filtered messages unless they wish to.
3. A real sender can easily get their message delivered directly to the users inbox by proving they are a human.
4. By identifying many different 'features' using different modules/methods the results are more robust than any single mechanism for identifying spam.
5. System admins can add their own 'features' and the system will correctly analyze the 'value' of those features so the systems reliability will not be degraded by 'one' badly chosen rule/score...

Filter Module Summaries:

- **SPF Domain Keys Sender Verification** - These systems allow the sender domain/ip address to be verified as genuine, although this doesn't stop all spam it can certainly stop a large bulk of spam and phishing by identifying the obvious forgeries.
- **Friends** - For each user we maintain a list of friends, messages from known friends are whitelisted automatically and delivered directly to the inbox, the list of friends is automatically updated by the server as it notices who you send email to, and what messages you move in/out of your spam folder.
- **Automatically collected sample database** - A sample database is automatically collected so that the filter rules can be tuned to match the types of spam and nonspam that are common on your server.
- **Bayesian like content filter** - Each day the database is used to re-tune the Bayesian like filter which works by statistically analyzing the words and order of words. This filter does well with text content spam (phishing, scams etc)
- **Phrase and Key Word Filter** - This filter is manually created and regularly updated to target phrases common to spam, this filter is also good at catching common scams and any common text based spam messages.
- **Web page probing** - This filter carefully examines messages to find url links which are 'unknown' to it, it then probes the web pages in question to see if they contain commonly spammed products (drugs etc)
- **SPF and Domain Keys sender verification** - Sender verification methods are used to identify people forging email, this allows non genuine email to be quickly identified and blocked. Most spammers forge email identities in some way.
- **Feature rule file** - This rule file identifies many different features of email (like capital letters in the subject), these features are then fed into the self tuning feature processing module...
- **Self tuning feature processing** - This module takes all the information from the modules above and combines the results based on analysis of the sample database to figure out the probability of a message being spam based on all the features it exhibits. Each day the rules are recreated based on your local spam samples, and combinational rules are created automatically.

Some more technical details

How sf_mfilter --> feature_gen.dat (note aspm_mfilter.txt is replaced with sf_mfilter.txt in this version)

The rule file `sf_mfilter.txt` now produces a list of significant features for any email message, the features are then analyzed using the rules in `feature_gen.dat` to come up with a 'score'. So the scores are not 'hard coded' into `sf_mfilter.txt`

The file `feature_gen.dat` is created by analyzing sample messages from your own server, so lets say we have a feature "blob" which on your server correlates 98% with spam, and on my server correlates 20% with spam (so in other words an email on your server with the feature 'blob' is a spam email 98 times out of a hundred, and on my server 80 times out of a hundred its not spam. Then on your server the score in x-spamdetect header will be something like "plus 10" for an email with 'blob' and on my server it will be 'minus 4'...

The feature 'blob' might relate to something like the length of the 'to' header, or weather or not the spf tests passed etc...

Then, in addition to simple rules the automatic process generates combinational rules based on your sample messages, so it might notice that a message which is from yahoo and has a long "To" header, is always spam. These 'combined' rules are also used to further increase accuracy.

Built-in RBL / Reputation system

SurgeMail now includes it's own RBL system (Realtime Blocking List) and Reputation system. This is a two level database, a local database based on each server, and a reporting system and DNS based query system to merge data between all SurgeMail servers in the world.

This system classifies all ip addresses into one of 5 colors

Unknown = 98% spam

Blue = less than 10 days old, nothing significant known (typically 70% spam)

Brown = 95% spam

Orange = 40% spam

Yellow = 20% spam

Black = 99% spam

White = Less than 4% spam

As most 'real' email comes from servers you talk to all the time this system quickly identifies the trusted mail servers that never send spam so that messages from those server will be very unlikely to be accidentally classified as spam.

The advantages this has over traditional RBL services (which should also be used of course)

It is free to use

It makes use of the many users clicking 'spam/not spam' on messages as they read them to help identify spam more accurately.

It provides both positive and negative indications for the spam filter, this is much more valuable than purely negative responses as given by many rbl services because the bulk of spam comes from 'unknown' transient ip addresses, so the significant information is the list of known mail servers that regularly send 'non' spam.

It is also a long term reputation rbl, so instead of automatically forgetting everything every 2 days like many rbl systems we try and store a long term record of stats for each ip address. This database can be searched here: <http://reputation-email.com/reputation/index.htm>

Management commands/settings etc

tellmail commands:

`tellmail sf_train` - Rebuild `feature_gen.dat` from `sf_mfilter.txt` using local data in 'train' subdirectories

`tellmail sf_compare` - Test `feature_gen.dat` on train sub directories.

`tellmail friends_url` - Show a sample URL for unblocking a message, use to test your web access/ports

are set correctly.

New Optional settings:

`g_myrb1_share "true"` - Share IP reputation information with netwinsite.com (**strongly recommended**, this setting really helps contribute to the wide area rbl which all customers benefit from)

`g_sf_generate "true"` - Generate `feature_gen.dat` locally rather than using a standard generic one from NetWin. This is worth setting once you have a reasonable sample collected (surgemail automatically collects sample messages within a few days)

`g_friends_lang_auto "true"` - Guess the users language(s) by observing messages from each users friends, then add a tag if the user receives a message which is primarily in a language that the user does not have listed. The users language settings are prefixed with the word 'Auto,' when this setting is used so users who have manually set their language(s) will not get adjusted.

Brief outline of how a message is processed

1. Get color from RBL/Myrbl/Surbl etc...
2. Run `sf_mfilter.txt` to find 'features' of message
3. Score message using `feature_gen.dat` and then bounce with url or give to user.
4. Run `mfilter.rul` file
5. If from friend accept
6. If exceeds friends setting then bounce message and store in 'spam' folder.
7. Deliver to inbox

Features to stop cracking local accounts and sending out spam

1. `g_breakin_enable "true"` - used to stop a spammer sending from multiple (3+) ip addresses. (`g_breakin_white` can be used in rare problem cases, e.g. `g_breakin_white "user1@domain.com,user2@domain2.com,*@domain3.com"`)
2. `g_user_send_warning` - alert manager when user sends too many messages.
3. `g_user_send_max max="500"` - limit users to a modest daily total
4. `g_safe_smtp "true"` - stops a user logging into surgemail to send email without first logging into imap or pop, this will stop 'most' spammers in their tracks even after they hack into an account (but not all) It won't usually cause people problems but it might on rare occasions.

Explanation of the X-SpamDetect header

Here is an example header:

```
*****: 7.8 sd=7.8 [194]99%13.1(!9,46) [126]10%-7.2(!33,108) [38]87%5.4(X-myrbl:unknown)"
```

This shows a score of 7.8, then a list of the rules that were applied seperated by spaces. There are two sorts of rules, simple rules and combination rules.

A combination rule looks like this: `[rulenumber]percent%score([!]a,[!]b)`

`rulenumber` = this rule number as listed in `feature_gen.dat`

`percent` = The percent of messages that have spam if this rule is true

`score` = The score which is generated using the percent. Anything over 50% generates a positive score, below 50% generatase a negative score.

`(a,b)` = The two rule which were true that made this combined rule true, ! signs are used to indicate 'not'.

A simple rule looks like this: `[38]87%5.4(X-myrbl:unknown)`

`rulenumber` = this rule number as listed in `feature_gen.dat`

`percent` = The percent of messages that have spam if this rule is true

`score` = The score which is generated using the percent. Anything over 50% generates a positive score, below 50% generatase a negative score.

`(a:b)` = The header and value that were 'matched' that made this rule true, if no header is specified

then it's a feature as defined in `spf_mfilter.txt`

The total `sd=7.8` is not a simple sum of rules, but rather an 'average' of the rules that matched. Offset by '4' to the right, e.g. `sum(scores)/n+4`

How to enable the new system (for those upgrading)

- Install Surgemail **5.0** or later
- Optionally set **`g_sf_generate "true"`** if you wish to generate your rule files from your own samples (which are automatically collected)
- If you are using friends we recommend you set friends to **level 6**
- Use **`tellmail friends_url`** to test if your external web address is accessible so a sender who is incorrectly blocked can bypass the problem.
- Convert your `local.rul` file (if you have one) to `sf_mfilter_local.txt`, see section below on `sf_mfilter_local.txt` (or try without any local extensions as you may find you don't need them)
- Global or user based `friend.msg` response templates are not used (as they would probably give the wrong instructions to confirm by email instead of using the url). So in the unlikely event that you need to tailor the friends response you (or any users) will need to re-tailor the message in the web interface as usual.

How to disable the new system (if you must)

To disable the new Smart Filter mechanisms and return to the old behaviour!

Only use these settings if you really must :-)

```
g_myrbldisable "true"
g_sfdisable "true"
g_friends_byemail "true"
g_spf_byemail "true"
```

How to TUNE it for your users/system (help I've got more spam or more bounces)

- Install Surgemail **5.0g3 or later and use the config checker!**
- Move 'webmail' users to 'SurgeWeb' (this gives easy access to the spam folder, and training buttons), encourage all users to try using `surgeweb` to read their email. To get to SurgeWeb they use a url like this: `http://your.mail.server/surgeweb`
- Make sure all users understand **how to get to their 'Spam' folder** (which will contain blocked messages!)
 - SurgeWeb users can see it automatically
 - IMAP users will see a folder called 'Spam', dropping message in/out of that folder will train the message as spam/not spam.
 - POP users are stuck, see details below.
- For POP users there are three choices.
 1. Don't filter their email just 'tag' it in the subject as normal. (This is safest)
 2. Set a default friends rule at about 9 or higher and set `G_USER_STATUS_SEND "1"` (units=days) which will send them a list of message in their 'spam' folder once a day. This is also a good option!
 3. Lastly encourage pop users to change to SurgeWeb or IMAP so they can use folders! (This is really the option when option '2' doesn't satisfy their needs)
- **HELP it's bouncing too much stuff** -- This probably means you or users have 'bounce' rules, upgrade to 5.0g and turn on the recommended features including `g_spam_nobounce "true"`
- **HELP I'm getting more spam now** -- This probably means you used to bounce email based on score (which was bad, naughty you! :-), now you have to turn on a global/domain/user rule for friends to have the same effect. If you are a 'pop' user (or most of your customers are) then set the level higher, e.g. '9' because as a pop user 'you or they' cannot see their spam folder to find false positives except by checking the daily status email!
- **Help local messages are marked as spam** -- This means you are missing these two settings: `g_smite_skip_relay "true"` and `g_smite_skip_auth "true"`

How to convert your local.rul file to sf_mfilter_local.txt

You can tailor your own rules still with this new system however, we suggest you consider the following, try using the builtin rules first and see how they perform.

If you are going to use 'local' rules then you must first enable local training:

g_sf_generate "true"

When adding rules (e.g. converting an existing local.rul file) you will need to change the actions to choose from the various possibilities

1. Add a manual score - call feature_manual(0.8, "Manual addition")
2. Add a self tuning score based on your spam sample - call feature_add(1.4,"featurename")

In the second case the score '1.4' is ignored. So the recommended method is to convert all call spamdetect(x,y) statements to call feature_add(x,y)

In the first case the value 0.8 is NOT the value added to the spam score, it is the probability that such a message is a spam message, so a value of 0.99 might add 12 to the spam score. A value above 0.5 will add a positive value to the spam score, a value below 0.5 will decrease the spam score. Examples:

```
call feature_manual(0.95, "Probably spam")
call feature_manual(0.7, "may be spam")
call feature_manual(0.0, "almost never spam")
call feature_manual(0.1, "Probably not spam")
```

You should only use 'manual' rules when the feature is so 'rare' that your sample data does not give useful figures on it, and in that case, the rule is probably of little or no value, so we suggest you don't use it at all :-). But there are exceptions where the sample spam messages will tend to give the wrong result (as the sample is not entirely random) so a manual rule might make sense.

A good example of when to use manual rules is when using an RBL service etc so you know that the probability of it being spam is very high you can improve results by using the manual rule with a nice high value like 0.9

Then run the following commands:

```
tellmail sf_train
```

```
tellmail sf_compare
```

The first will generate a feature_gen.dat rule file and the second will use it to compare results with the sample spam folders.

If you examine 'feature_gen.dat' after the sf_train command you will be able to see what surgemail thought the feature was and how significant it was (sig = the number of messages with the feature), A feature with a probability near 0.5, or one that occurs less than 20 times in the sample is probably of little point... Near 0.0 means the feature implies the message is not spam, near 1.0 implies the feature correlates with spam...

We are always interested in new features you make up that prove useful. Be cautious that some features can give misleading results due to the nature of the sample messages.

Convert into automatic rules instructions (recommended)

- copy local.rul to sf_mfilter_local.txt

Search/replace "spamdetect(" with "feature_add("

Convert into manual rules instructions:

- copy local.rul to sf_mfilter_local.txt
- Search/replace "spamdetect(" with "feature_manual("
- Change each spam score -x to +x into a probability instead (0.0 -> 1.0) of spam.

Tips for users to avoid spam:

Never put your email address on a web page, instead use a service like this one: <http://www.emailmeform.com/>

What if it doesn't work at all ?

If the scoring is completely blank or if you see this text in the headers:

X-SpamDetect: : 0.0 sd=0 feature_gen.net (or .dat) is blank or missing, update from netwinsite failed see netwinsite.com/surgemail/help/myrbl.htm for help

It might mean you are running a new build with the new spam handling mechanism, and most likely it's failed to pickup it's main rule file so it's not applying any rules at all.

It might fail if you don't have updates, or if you have a firewall blocking port 80 outgoing connections from your server. Once you fix the problem you can 'trigger' an update automatically by deleting `aspm_update.done` and restarting surgemail.

The two files you need are:

sf_mfilter.txt
feature_gen.net

They should automatically be fetched from netwinsite but that 'can' fail if your firewall is blocking port 80 connections. In which case you could download them manually then restart surgemail.

http://netwinsite.com/surgemail/sf_mfilter.txt
http://netwinsite.com/surgemail/feature_gen.net

Or You can disable the new system with this setting: **g_sf_disable "true"**

Like

2 people like this. [Sign Up](#) to see what your friends like.

SurgeMail Legal Archive

To use the S3 service you will first need to [Register](#) at amazon, then to activate your S3 subscription in SurgeMail paste the key into the web admin interface, click on "Legal Archive" on the left hand side of the web admin tool and you will see a box to enter it.

Summary

The Legal Archive feature in Surgemail is intended for storing a copy of all incoming and outgoing email for several years. The archive can be searched by the administrator or directly by users (this can be disabled).

Messages are compressed, encrypted, and indexed, then optionally sent to the Amazon S3 storage network for redundant offsite storage without over loading your local disks.

When a user requests a mesesage just that message is fetched and decrypted so the message is available instantly. The indexes are stored locally so searching for messages is relatively fast.

Settings (To adjust settings use your web admin interface)

Essential Settings:

Enable Legal Archive	
Path for indexes (this may use significant space)	
Encryption Password (if you loose this all your archives will be permantently lost - write it down NOW)	

Optional Obscure settings, best left alone :-)

Disable Amazon/S3 (Keep files on local disks)	
Bucket to store data (if you change this all existing archives will be invisible)	
Host number (for shared storage clusters, to give each host a unique identifier on amazon) 1...n	
Years to keep messages (default 7)	
Limit direct search of archived messages to those in the archive group	

Amazon-SES

This guide helps you setup SurgeMail to send outgoing mail via an Amazon-SES account.

Signup on amazon's website: <http://aws.amazon.com/ses/>

Follow this step by step guide to get your identifiers and install the developer tools perl scripts and to verify your account:

<http://docs.amazonwebservices.com/ses/latest/GettingStartedGuide/>

You should now have the perl scripts including: ses-send-email.pl copy this perl script to send.pl and add a line near the top so it will work when run from another directory, lets assume you installed it in c:\surgemail\amazon

Then you would add:

```
use lib '/surgemail/amazon';
```

Next add a setting to surgemail to redirect outgoing email to this script, on windows the setting would look like this if you placed your credentials in a file 'cred.txt' in that same directory...

In this sample rule you should just use the email address you have verified, once tested you can replace it with "*" to use this robot for all outgoing email.

```
g_redirect_ses was="VERIFIED@ADDRESS.COM" to="|\perl\bin\perl.exe \surgemail\amazon\send.pl -r -k \surgemail\amazon\cred.txt"
```

On unix the slashes would go the other way and \perl\bin\perl.exe would probably not be required.

Now send a test message.

If it fails examine the logs, find the robot command line used and test it manually to see if it works.

SurgeMail Change History (older)

For recent release versions of surgemail see the [recent surgemail change history page](#).

SurgeMail 1.8g3 25-March-2004

- Spool path allowing SurgeMail to send dropped files as email (g_spool_path)
- New settings (g_delete_user_mode, g_user_alias_file)
- SurgePlus calendar and filesharing (fully functional but still beta)
- Fix: Memory corruption bug causing sporadic random crashes on SSL connections
- Fix: Vpipe restart bug on timeout
- Fix: Further fix to new na_exceptions web page
- Fix: Minor memory leak
- Fix: Proxy mode fix
- Fix (1.8g2): WebMail autologin fix for non default vdomains using custom quick login method
- Fix (1.8g2): Alias creation bug
- Fix (1.8g3): Hopefully final fix to WebMail autologin. (Rolled back logic to that of previous release builds)
- Fix (1.8g3): Reports page broken on UNIX systems
- Fix (1.8g3): Bug in SurgePlus that could sporadically crash SurgeMail on display of SurgePlus web pages

SurgeMail 1.8e 11-March-2004

- Additional "build number" added to version information to uniquely identify surgemail builds
- # character on first line of surgemail (not webmail) templates means comment
- Unmonitorable mail blocked by Avast by default (to allow use g_virus_allow_unmonitorable)
- Fix: Delete user removes the user from all mailing lists
- Fix: Mirror tidyup in logging of email actions in users log files

SurgeMail 1.8d 9-March-2004

- Web based Queue handling / delete / retry etc
- User.cgi exceptions forward and bounce with reason ability
- Additional old_pophost migration setting (g_fallback_relay_if_exists)
- Fix: Sporadic memory corruption crashes
- Fix: Global / domain exceptions actually listed in order processed
- Fix: Autogenerated g_server_name entries based on url_host no longer overwrite existing entries

SurgeMail 1.8b3 4-March-2004

- Fix: Crash on aspm message training
- Fix (1.8b2): User.cgi autologin broken (fixed again)
- Fix (1.8b2): Aspm URL scoring was broken
- Fix (1.8b2): Aspm now installed as default antispam scanner (as opposed to SmiteCRC - which is still fully supported)
- Fix (1.8b3): Rollback pre-release version of na_exceptions.htm template

SurgeMail 1.8a 25-February-2004

- Enhanced (and much faster) Report page
- Allow the use of return codes from g_virus_cmd virus scanners (g_virus_cmd_codes)
- Options to control the content of your periodic account log message.
- New version available indicator on status page
- Aspm test and training page in web admin interface
- SMTP verify option (g_verify_smtp)
- Searchable and sorted by date friends pending / spam pending etc lists
- Allow relay based on authent return code (lookup_relay_on_from)
- Dumping of pstat statistics to text file "tellmail pstat_dump" (for import to excel)
- Improved slightly odd behavior in places of web admin interface
- Fix: Browser image caching fixed - makes webmail use much faster!! :-)

- Fix: Allow SSL certificate creation to work with Microsoft CA
- Fix: Migration bug that resulted in duplicate messages and failure to create account if first login was IMAP and account was being migrated using old_pophost
- Fix: Migration bug that allowed IMAP migration to lose all message flags
- Fix: Bottom left hand navigation pane links not working if clicked while surgemail was not running (IE only)
- Fix: Several minor friends confirmation message fixes and enhancements
- Fix: Timezone information displayed incorrectly in received headers in 0 to +9 timezone
- Fix: IMAP fixes
- Fix: Improved shutdown behaviour
- Fix: High loading mutex timeout crash.

SurgeMail 1.7b3 4-February-2004

- Fix: Aspm functionality improved
- Fix: Mirroring crash bug (introduced in 1.7a)
- Fix: Solaris_x86 only crash bug when doing log file searches
- Fix: Installer waits until surgemail has fully shutdown (was sometimes failing file copy on first upgrade attempt)
- Fix (1.7b2): User.cgi blank pages displayed for non admin users (introduced 1.7a)
- Fix (1.7b3): Aspm file update, crash and occasional incorrect match

SurgeMail 1.7a 27-January-2004

- ASpm more accurate and efficient antispam system
- New more accurate and efficient quota handling
- Ability to limit bounce message size (g_bounce_limit)
- Integrated fprot antivirus support (g_virus_fprot)
- Option to have spam store / friends pending folder part of / not part of quota (g_xxx)
- IMAP folder renaming also renames subfolders
- Mailing list administration for domain administrators
- User domain defaults administration for domain administrators
- Fix: Mirroring of dlist subscribe.lst file

SurgeMail 1.6e2 9-January-2004

- Fix: Memory leak in sending of friends status messages
- Fix: Deletion of some necessary files in the surgemail directory (only happens under certain circumstances and only affects unix versions, if nauth is used and mirroring is enabled)
- Fix: Account "Access Group" information no longer applied if no g_acces_group rules defined
- Fix: Improved handling of mirroring (in particular dlist subscribe.lst mirroring)
- Fix: Improved handling of very large numbers of messages in inbox (30000+ messages)
- Fix: Improved handling of friends bounce messages

SurgeMail 1.6e 23-December-2003

- Domain specific footer file based on from envelope (footer_file)
- Account creation check disable when using old_pophost / old_imaphost (old_smtphost, old_smtphost_skip)
- Fix: Linux memory fragmentation under heavy load fix
- Fix: Sporadic surgemail crash in user cgi processing

SurgeMail 1.6d 17-December-2003

- Allow intercept migration mail to be left on old server (old_pophost_nodelete, old_imaphost_nodelete)
- Per user enabling and serverwide configurable subject marking of spam (g_spam_subject_word)
- Latest version of webmail (version v3.1d build 6)
- User account aliases with quotas (g_user_alias, g_user_alias_file)
- Per user quotas for sending sms messages
- Fix: Sporadic web page corruption (particularly noticable in navigation pane)
- Fix: Several cases where intercept migration failed prematurely (eg on messages with no body)
- Fix: Save on domains page takes long with high cpu usage (surgemail/web_work/surgehost.ini was getting excessively large)

SurgeMail 1.6b 28-November-2003

- Allow users to import all mail from external POP / IMAP accounts
- Optional automatic domain dropdown for webmail and user cgi (g_user_domainlist, domain_select)
- New version of webmail (3.1c build 11) with simplified options pages in panel template set
- Fix: Quota in quota.txt file drift issue fixed
- Fix: Webmail panel template horizontal alignment issue in Opera browser
- Fix: Some minor imap issues fixed

SurgeMail 1.6a 19-November-2003

- Configurable signup emails (signup_user.eml and signup_manager.eml)
- Allow users to create aliases (g_user_alias_file, g_user_alias, user_alias)
- Additional spam control settings (g_black_*, g_verify_*, g_spam_subject_gateway, g_spam_allow_recent, g_spam_check_auth, g_from_bounce, g_from_stamp,)
- Minor changes in account status / spam hold mechanism and notification emails
- Additional gateway settings (g_gateway_always, g_gateway_ifnot)
- Run late mfilter at local delivery time (g_user_mfilter)
- New version of webmail (3.1b build 45)
- New miscellaneous settings (g_from_timeout, g_filter_max, g_drop_use_len, g_filter_max, g_create_allow_pass)
- Allow footers to be attached to outbound messages only (g_footer_send)
- Fix: Web admin interface javascript issues on OSX Safari and IE browsers
- Fix: IMAP intercept mode upgrade some fixes (of note on a few mailservers upgrade would complete if empty folder encountered - Mailsite in particular)
- Fix: IMAP OE fix that would display error if folder was emptied by another mail client
- Fix: Windows98 slow response time issue fixed
- Fix: Self signed certificates now work on OSX browsers - certificates need to be regenerated

SurgeMail 1.5f 31-October-2003

- Global and per domain defaults user setting now available
- Redesigned status summary information (includes a variety of additional information such as migrated users, free disk space, etc)
- Advanced log file searching capabilities
- Support for intermediate SSL certificates (eg. as issued by COMODO)
- Certificate Signing Request generation / certificate management through admin web interface
- Easier tracking of spam bounces (Msg.rec contain spam ratings, bounce message contains queue id)
- Ability to set retry hours per domain (g_retry_rules)
- Delivery time filter like g_filter_pipe (g_user_pipe)
- Improved surgemail catching of messages that are not correctly processed by AVAST
- Fix: Queue file backlogs would not be properly processed sometimes
- Fix: Mirroring now smarter in the way it propagates updates
- Fix: User cgi pages fail to autologin back to webmail
- Fix: Correct determination of system directory on Windows 2003 multiuser systems
- Fix: Display of pending stored messages due to filename truncation
- Fix: Correct handling of the avast installation dll during surgemail upgrades
- Fix: IMAP folders staying locked if using the BAT mail client with IMAP

SurgeMail 1.5d2 9-October-2003

- Latest SSL libraries
- Fix: IMAP memory leak
- Fix: Duplicate UIDL fix

SurgeMail 1.5d 1-October-2003

- Ability to set log file size (g_log_size)
- Ability to specify AVAST update times (g_virus_avast_hour)
- Fix: Security fix

- Fix: Added workaround to fix IE status bar failing to recognise full document is downloaded (sometimes)
- Fix: Improved recovery of AVAST failing to update engine and spam database
- Fix: Web admin save of domain settings save lost surgewall setting
- Fix: Minor IMAP fix of specific attachment corruption
- Fix: Bulletins not delivered to subdomain accounts

SurgeMail 1.5c 24-September-2003

- Ability to specify specific ip's and ports for surgewall mode
- Fix: 100% CPU loop that occurred if one of several redirect_cc's failed (introduced post 1.4b)
- Fix: Status page message count "unsent yet" inaccuracy
- Fix: OSX mail client was not displaying some IMAP messages properly
- Fix: IMAP proxying in surgewall mode
- Fix: default domain quota was being used for vdomains

SurgeMail 1.5b 19-September-2003

- Quite a few changes in the web admin user administration pages
- Exception rules are now external to Friends and Spam system
- Latest version of webmail (version 3.1a build 8)
- Access defaults for users not in any g_access_groups (user_access_default, g_user_access_default)
- Smitecrc is more efficient in memory use
- Domain admins can now send bulletins
- Fix: Automatic correction to surgemail key login if smitecrc smitespam login details are incorrect
- Fix: Several crashing bugs removed (primarily solaris affected)

SurgeMail 1.5a 9-September-2003

- Vipe now autodisable vpipe if it has three errors in a row (unless g_virus_filter_require is set to true)
- Add spam headers to body changed from boolean to numeric spamdetect rating
- Add more options to the Friends/Spam exception rules (still in friends page - will be moved from here)
- Modified the way the surgemail release [numbering scheme](#) is used (no functional surgemail changes)
- Ability to limit the max number of emails a user may send in 30 minute period (send_limit=n in authent response)
- Fix: Report page broken on unix platforms on the first 9 days of the month
- Fix: File permissions issue on upgrade - primarily OSX but could affect other unix platforms
- Fix: Access permissions in "Access type" were overridden by "Account Status"
- Fix: Webmail now provides more informative error messages if the login fails due to access limits
- Fix: If friends rule had no * or ?, and had a trailing space rules would incorrectly match
- Fix: Empty folders created properly for old_imaphost migration + migration on linux fixed
- Fix: Several fixes in smitecrc and the way smitecrc is run
- Fix: Bounce messages if workarea full - previously under some conditions blank messages were sent
- Fix: Fixed error handling in surgemail if AVAST setiface.dll is corrupted (would deadlock surgemail)

SurgeMail 1.4c 2-September-2003

- CentiPaid micro payment system.
- Web admin now "remembers" domain you are working on.
- Low disk warning notifying admin if work paths or mail delivery paths fall below this level (g_low_disk)
- Auto responder options to respond always/once a month/once ever to each user.
- Disable accounts / delete user after period of inactivity (g_disable_smtp_after, g_delete_user_after)
- Cookies allowing automatic login to web admin interface
- Web admin asks for confirmation before deleting domains and users
- Fix: Domain admin redirect + redirect_cc pages allow one of multiple domains to be selected and fixed surgemail crash
- Fix: Various fixes to old_imaphost intercept mode migration (was significantly broken)
- Fix: Search for particular email in webmail
- Fix: Bug in spam held messages that meant that no status message was being sent out and old spam was not being purged
- Fix: Bug in friends systems that meant the friends report email continued to grow
- Fix: DNS failover to other dns server if one is broken (was not working correctly)

- Fix: Web admin now shows system/domain administrators stored messages rather than their own.
- Fix: Solaris only issue that ?????? was being sent as from/to addresses when using IMAP.
- Fix: Several problems running surgemail on Win 9x systems.
- Fix: Path issue with dlist
- Fix: Several instability issues fixed

SurgeMail 1.4b 15-July-2003

- Latest version of webmail including updated spam settings templates
- Added DNS translation (g_dns_translate)
- Improved per channel connection statistics
- Fix: Several AVAST related bug fixes
- Fix: Minor friends system bug fixes
- Fix: Minor user spam system bug fixes
- Fix: Web admin UI bug introduced 1.4a

SurgeMail 1.4a 30-June-2003

- Installer rollback functionality (run "surgemail -rollback" on command line)
- AVAST antivirus integration (windows only and needs to be purchased separately)
- Improvements to surgewall
- Per user virus / spam / other filter enabling and disabling
- Allow quota warnings to be disabled (g_quota_warning_disable)
- Forward and reverse dns lookup check (g_dns_paranoid)
- Additional gateway features (g_auth_skipgateway, g_gateway_auth, g_local_skipgateway)
- Additional spam settings (g_spam_vanish, g_spam_vanish_all - similar to g_spam_bounce settings)
- Early (prior to filters) message archival (g_archive_early)
- Fix: Account status bug fixed (introduced 1.3d)
- Fix: Drift in quota processing fixed
- Fix: Per user spam filtering was badly broken
- Fix: Several Friends fixes
- Fix: Variety of minor fixes

SurgeMail 1.3l 10-June-2003

- New webmail version (3.0x)
- Addition of g_authent_always, g_domain_default, g_friends_confirm_subject settings.
- Fix: Domain web admin page has all missing settings (surgewall and old_imaphost*)
- Fix: "Delete original message" disabled by default when forwarding / using responder
- Fix: Several of IMAP protocol RFC compliance fixes

SurgeMail 1.3k 30-May-2003

- Surgewall feature - the ability to place surgemail as a filter "in front" of an existing mailserver to apply friends rules, spam filtering, virus scanning (see domain setting surgewall)
- IMAP intercept mode migration (see old_imaphost_*)
- Friends settings improved
- Mailbox manipulation rules in user account management
- Domain admins can modify users' friends and spam settings
- Stop accepting mail for accounts that have not been accessed for certain time (g_disable_smtp_after)
- Allow retry period to be set (g_retry_bounces)
- Improved handling of excessive invalid logins
- All multivalue settings are now ordered
- Fix: Bug that messages were sometimes bounced if received within 1 minute of pop login (introduced 1.3j)
- Fix: IMAP Allow subfolders in Outlook Express + several minor folder update issues
- Fix: minor IMAP quota bug
- Fix: Bug that first Friends email was sometimes not delivered

SurgeMail 1.3j 10-May-2003

- Ability to search for administration settings

- Allow old_pophost to be checked subsequent to first login (old_pophost_always)
- Additional control over user cache (g_authent_cache size, g_authent_cachebad)
- New tarpit setting (g_tarpit_badrcpt)
- Tailorable Quota 80%, Quota reached, and bounce messages (warning.eml, quota.eml, failed.eml files)
- Automatic daylight savings timezone identification for "Received" headers
- Web admin interface changes to more effectively handle large numbers of domains
- Fix: Account quota file gets automatically rebuilt to prevent incorrect quota limit messages
- Fix: Automatically removes trailing spaces from 'mail from' envelopes
- Fix: Additional header linefeed(if headers over 2kB and packet broken on cr/lf boundary)
- Fix: Variety of minor fixes

SurgeMail 1.3i 30-April-2003

- Ability to search for administration settings
- Display users' quota in administration interface
- Allow virus scanning to be enabled on per user basis (g_user_virus_scan)
- Allow server side storing of suspected spam messages (see user's spam settings)
- Allow url aliases (url_alias)
- Fix: Under certain conditions swatch would not restart surgemail
- Fix: Add missing contents template files

SurgeMail 1.3h 28-April-2003

- Fix: Bug in g_ssl_per_domain preventing surgemail startup if set.

SurgeMail 1.3g 24-April-2003

- Frame driven administration interface
- Added quota warning message at 80% quota and limit message at 100% quota
- Custom smitecrc spam detect filters using local.rul file (see filter.dat for rule syntax)
- Addition of arbitrary message scanning (g_scan_*) (equivalent of dmail process command)
- Addition of cc forward based on from address (g_forward_from_cc)
- Valid recipient check on from envelope (g_badfrom_check, g_badfrom_stamp)
- Spambot is now part of distribution
- Fix: Several friends system fixes + features (add g_friends_ignore, only new entries in report, 1st email delivery correct for automatically add new addresses)
- Fix: Corruption of webmail surghost.ini when settings of existing domain saved in surgemail admin interface
- Fix: Timezone in Received header changes correctly with daylight savings time
- Fix: Trailing spaces in delivery address stripped before attempted delivery
- Fix: Memory leak
- Fix: Virus command (g_virus_cmd) crash on bounce containing virus and certain attachments not being deleted

SurgeMail 1.3f 2-April-2003

- Fix: Some user account management actions causing crash (introduced for 1.3e)
- Fix: Friends message uses correct email address in text
- Fix: Sporadically swatch could fail to restart surgemail (windows only)
- Fix: Smitecrc excessive CPU usage (introduced for 1.3e)

SurgeMail 1.3e 28-Mar-2003

- Force smite headers to be added to all mail going through server (g_smite_all)
- Allow custom smite filter rules through the use of #include "local.rul" in filter.dat.
- Allow sender ip to be specified in reported ORBS stamp messages (use ||remoteip||)
- Authent cache lifetime setting (g_authent_cachelife)
- Manager approved account create without further user validation (when users do not have prior email address)
- Smart updating of surgemail administration web templates
- New version of Webmail (3.0u build 25)
- Fix: Installation of missing webmail executable (introduced for 1.3d)

- Fix: Conversion of drop files was losing first line of headers
- Fix: Sporadic webmail CPU loop

SurgeMail 1.3d 20-Mar-2003

- Per user options for SmiteSpam system
- Addition of original recipient and authenticated user headers
- Allow smite headers to be added to gatewayed mail
- Allow old_pophost to be used to retrieve mail if user already in local database (old_pophost_iffirst)
- Allow webmail to autologin when running on a different server from surgmail
- Allow exceptions to g_con_perip (g_con_perip_except)
- Allow access group membership based on arbitrary authentication module fields (g_group_field)
- Allow mfilter to be applied to local files only (g_mfilter_localonly)
- Web UI tidyup
- Fix: Freebsd stack overflow problems (random crashes)
- Fix: Fixes to the IP failover system
- Fix: Incorrect date display
- Fix: Run mfilter over pop_fetched mail
- Fix: Improved log file handling
- Fix: Removal of several obscure bugs that could cause surgmail crashes and apparent lockups

SurgeMail 1.3c 18-Feb-2003

- Queue analysis
- Improved smitecrc integration
- New version of webmail
- Mfilter rules reloaded with test command or tellmail reload
- Fix: DNS issues that were slowing large systems
- Fix: Variety of stability fixes

SurgeMail 1.3b 24-Jan-2003

- Addition of multiple orbs services (g_orbs_list)
- Can force all users to use ssl but still allow plain smtp delivery (g_ssl_require_login)
- Ability to disable primary web interface
- Added IMAP SSL capability (dedicated port + STARTTLS)
- Easier configuration if host name is not the same as domain (eg host = mail.mydomain.com)
- Fix: Attachment matching in dmail dfilters
- Fix: Crash on several strange message syntax issues

SurgeMail 1.3a_rc1 14-Jan-2003 (Production release 1.3a on 24-Jan-2003)

- Extensive server upgrade and migration support
- Ability to block and name translate attachments
- mfilter improved logging and fixed several minor bugs
- Fix: IMAP folders properly displayed by MS Outlook
- Fix: Variety of minor issues

SurgeMail 1.2c 3-Dec-2002

- Allow non local addresses be added to mailing list using web admin interface
- Use url_host (if defined) in per vdomain ssl certificates
- Fix: Problems with subdomain delivery
- Fix: Variety of minor issues including
 - Links in domains page of admin interface
 - Single link in account creation confirmation email
 - Keep correct access privileges when resetting administration password

SurgeMail 1.2b 13-Nov-2002

- Improved delivery log file flexibility (see `g_log_path`, `g_record_path`, `g_record_days`)
- Improved Friends system: Customisable naming, customisable kids-only mail, fixed several bugs
- Skip virus & crc check (by IP) for known safe bulk mailouts (`g_vpipe_skip`)
- Additional spam prevention settings (`g_tarpit_blackhole`, `g_con_persubnet`, `g_tarpit_drop`, `g_tarpit_max`, `g_tarpit_max_remote`)
- Automatic SSL certificate generation on demand, and per vdomain SSL certificates
- Web admin : feature additions and minor fixes
- Allow monitor to be disabled
- Allow mail server name to be different from domain name for user account management (`g_server_name`, `url_host`)
- Upgrade installs will optionally update surgemail and webmail template files
- Fix: Possible message loss if `g_virus_cmd` virus checker is used (if you use `g_virus_cmd` you MUST upgrade immediately)
- Fix: Several user account management fixes including admin password on mail verification, list mailbox
- Fix: Several bugs in web based mailing lists administration

SurgeMail 1.2a 22-Oct-2002 (only released as beta)

- Allow individual surgemail subsystems to be disabled
- Webmail license key integrated into surgemail
- Attachment filename translation
- Tidier ini file handling
- New version of Webmail
- Fix: Show messages not displayed by some imap clients
- Fix: Handle leak under certain conditions if user over quota
- Fix: Hide surgemail path in SMTP error on user over quota
- Fix: Only output over quota message a single time in logs
- Fix: Minor web admin changes

SurgeMail 1.1d 11-Oct-2002

- Fix: Issue adding friends to friends system
- Fix: Sporadic ini file corruption upon web based administration (introduced version 1.0d)

SurgeMail 1.1c 7-Oct-2002

- Fix: subdomain delivery loop (introduced in version 1.1a)

SurgeMail 1.1b 4-Oct-2002

- Fix setting of authentication modules keeps settings
- Fix IMAP timeout when sending in mail
- Fix broken Report Refresh
- Fix broken delete accounts in NWAuth

SurgeMail 1.1a 30-Sep-2002

- Complete support for Friends + SmiteSpam in web admin UI
- Extended reporting facility
- Addition of startstop.log file
- More comprehensive manual
- Improved install + upgrading + uninstall
- Improved server mirroring
- NetAuth enhancements
- Fix: Login with failed password cached making local delivery fail
- Fix: Swatch settings changes does not require surgemail restart
- Fix: Sporadic NetAuth password Q&A field corruption
- Fix: Sporadic Surgemail crash upon IMAP access
- Variety of minor bug fixes and enhancements

SurgeMail 1.0d 9-Sep-2002

- Integration of SmiteSpam antispam system
- Integration of Friends antispam system
- Automatic identification of domains for user account administration
- Group based access control to POP, IMAP,SMTP
- Account status control
- Archiving of all received messages
- Addition of several new configuration options
- FreeBSD webmail issue, Windows tellmail issue
- Continued minor bug fixing

SurgeMail 1.0c 15-Aug-2002

- First production release build
- Updates have been made to many areas recently, so please upgrade to 1.0c at a convenient time as a baseline. Recent changes include:
- Help system now updated
- OpenSSL integration complete for all platforms
- Fixes in mirroring code
- Fixes in SpamAssassin integration (g_filter_pipe)
- High end server proxy mode support

SurgeMail CalDAV User Setup

This provides surgemail standalone CalDAV calendaring support (including calendar sharing) for mobile devices and desktop clients.

If it needs to be specified the full caldav url is:

<http://yourserver.com/cal/calendars/email@domain.com>

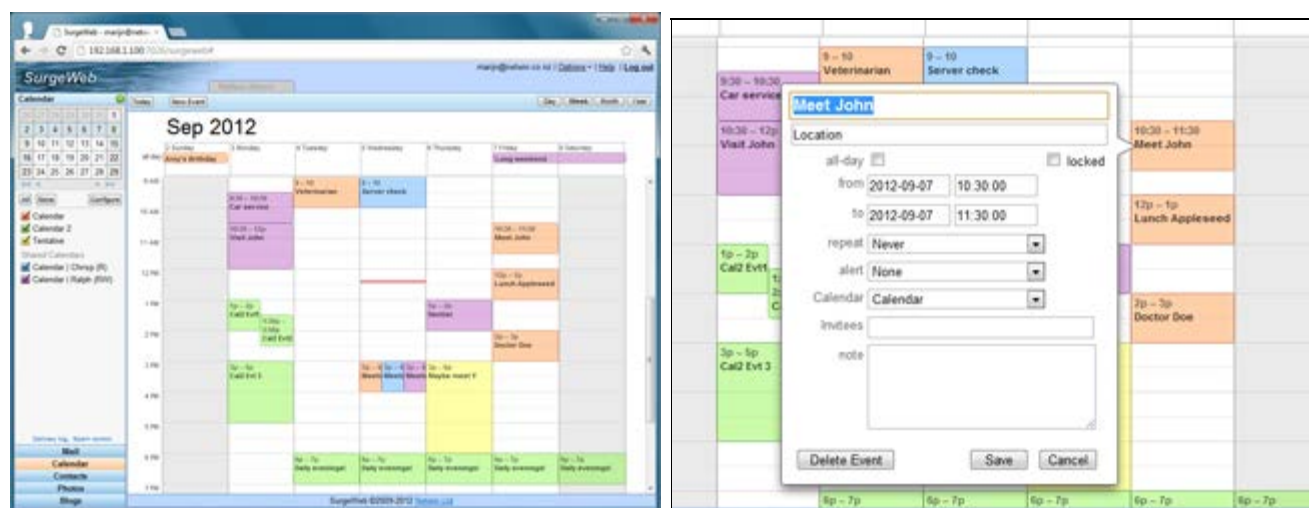
Username needs to be full **email@domain.com** and password is your normal email account password.
Configuration walkthrough of some sample clients below:

SurgeWeb as a CalDav calendar client

SurgeWeb now supports CalDav integrated "drag and drop" calendaring from the surgeweb ajax interface. Provided this is enabled on surgeweb customisation page, no surgeweb configuration is needed :-)

Key features include:

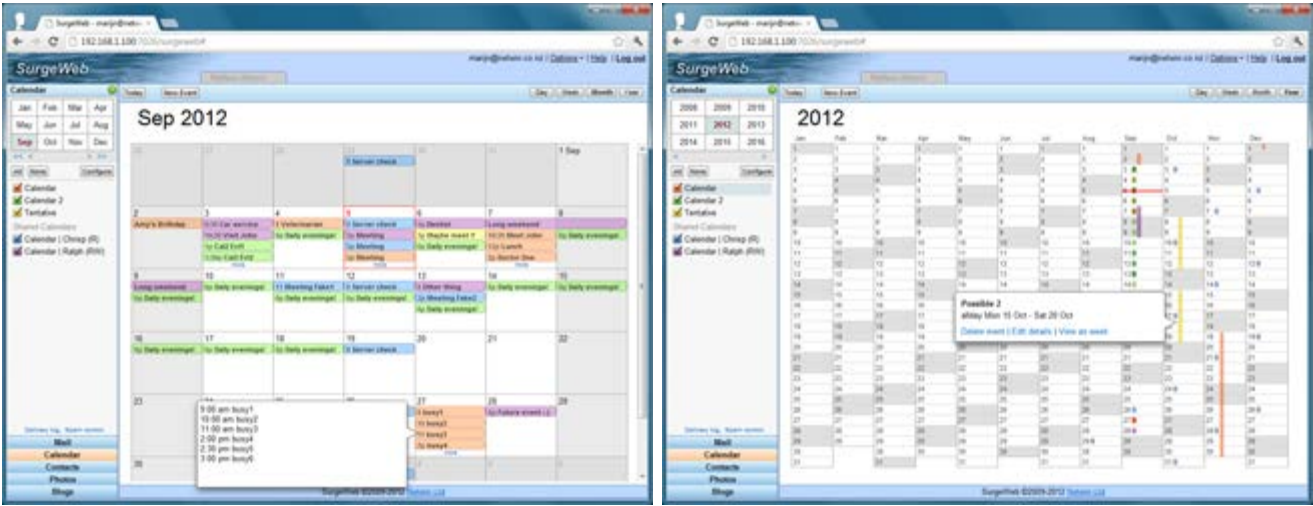
- Calendar synchronisation with mobile clients
- Calendar sharing between mutiple accounts
- Define multiple additional calendars
- Configure alerts that will trigger on mobile devices
- Repeat event support



(click screenshots to enlarge)

Additional features include:

- Support for: Day, week, month and year views
- Customise display color per calendar
- Point & click / drag & drop event manipulation
- Single key stroke switching between: date range and scale (one or more modifier keys plus arrows)

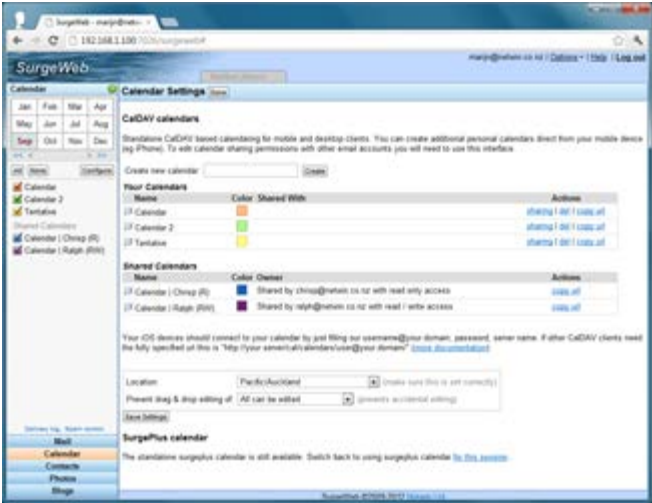


Configure Calendar Sharing using SurgeWeb

Surgemail CalDAV calendaring will allow you to specify one or more calendars for each email account. Each of these calendars may be shared with other caldav users on the same system. Sharing is configured using the surgeweb interface (surgeplus calendaring page).

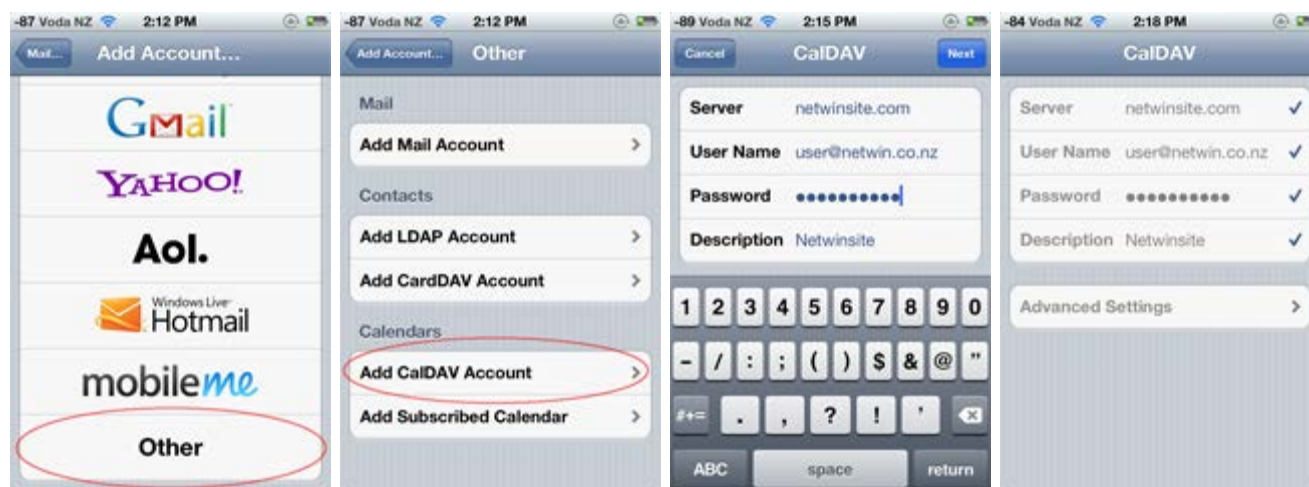
Calendars can be shared with the following permissions, and displayed to the person you are sharing it with as:

- read / write access: displayed as "Calendar Name | Owner (RW)"
- read access: displayed as "Calendar Name | Owner (R)"
- free / busy access: displayed as "Calendar Name | Owner (B)"

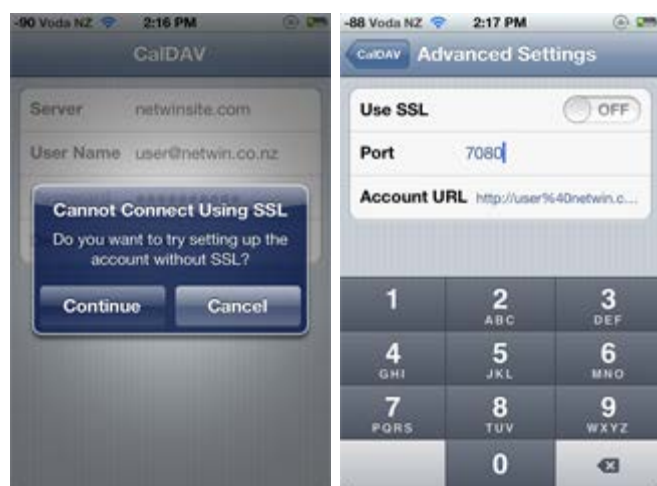


iOS Client Configuration

Standard account configuration under iOS should simply be a case of adding a new calDAV account and correctly filling out the server, fully specified email address as user name and the password:



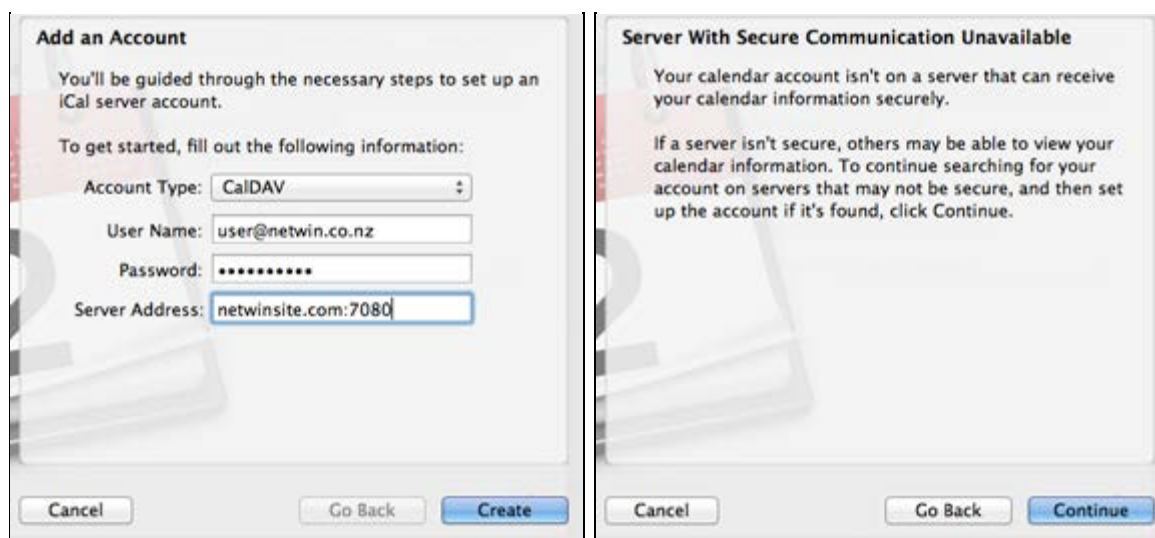
In addition you may need to accept non ssl based connections if you do not have a CA signed SSL certificate and specify the server port if this is non standard :



iOS will automatically detect the full caldav url, and automatically detect any calenders you have already got defined under the server on your account. Any calendars others have shared with you will also be automatically detected and displayed in your list of calendars.

OSX iCal Desktop Client Configuration

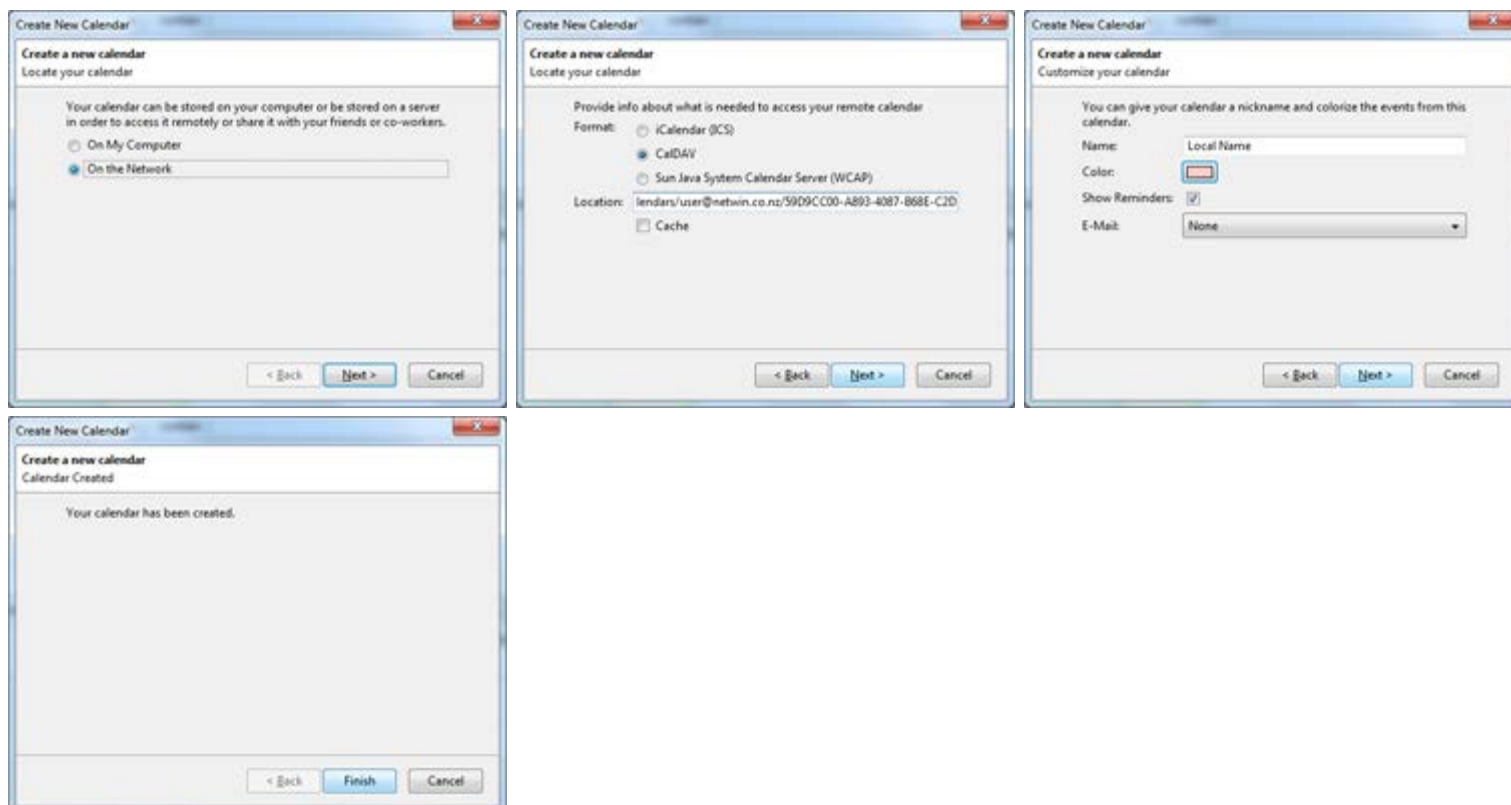
iCal will also automatically detect the full caldav url and all calendars based on valid username, password and server address settings. Again you may need to confirm acceptance of non ssl based connections if the server does not have a valid SSL certificate setup.



Windows Lightning Thunderbird Extension Configuration

Lightning has been tested and seems to work well as a free windows desktop CalDAV client with surgemail CalDAV calendaring. Setup is slightly more involved as it does not seem to autodetect the caldav url or calendars. (there may be ways around this - let me know if you find them)

To connect to existing calendars, use the surgeweb interface to copy the full url of any of your available calendars (or create / share calendars as needed), then add this as a new network calendar in Lightning:



Android calendaring client

Android "CalDAV-Sync" (no endorsement for this particular product and netwin is not affiliated with the developer in any way) has been tested to successfully allow surgemail CalDAV calendars to be used using the native android calendar client.

To configure, just login using the url of `http://yourserver.com:port/cal` as the url and email address and password

and all existing calendars should get detected.

Other CalDAV Clients

A variety of other CalDAV clients are available and may suit your needs better than the above listed clients.

For administrators also see caldav [server configuration](#).

Multilanguage support

Surgeweb now has extensive multilanguage support throughout the interface (as per [help page](#)).

This is a list of mainly customer contributed SurgeWeb translations. These are not guaranteed to be perfect or fully uptodate with the latest version of surgeweb, but should act as a good starting point for your own translation.

Customer contributed SurgeWeb translations

Danish: surgeweb [lang_danish.dat](#), user.cgi [lang_web.dat](#)
contributed by Claus Michael Holm, 5 November 2010

Please contact surgeweb-support@netwinsite.com if you have a translation file you would like to contribute :-)

Thanks!

SurgeWeb bug fix specifics

Spam Reporting issues (Surgeweb and IMAP) (fixed version 4.3f-13)

Complete rewrite of the way messages are marked spam / not spam from surgeweb fixing a bunch of oddities and making the code a lot more maintainable:

Previously:

1) if `g_spam_folders` or `g_spam_folders_show` was not specified

- Surgeweb "not spam" in the spam folder would subject clean the selected messages, and move these to the inbox. This would not release additional messages from the same sender, and this kind of "spam release" would not work with late forwarding / surgewalled accounts (nor would the "allow once" action)
- IMAP moving messages to the inbox would notspam train the individual messages, but not release additional messages.

2) if `g_spam_folders` or `g_spam_folders_show` was specified

- Surgeweb "not spam" in the spam folder would release the selected message and all messages from this user (in a way compatible with late forwarding and without cleaning the subject).

In addition, if this message was not the top message in the list:

- it would also display an error about "invalid copy parameters".

If the original message was the top message in the list:

- it would also release the second message on the list and any additional messages by the same sender. As well as adding a duplicate of the second message to the surgeweb inbox, with the subject cleaned.
- it would delete the third message in the list.

Also the surgeweb displayed message list would not get updated with the additional message releases & deletes, for the next 15 seconds even folder clicks would not update the message list in the spam folder correctly (unless a refresh folder was explicitly done, in which case it did display the modified Spam folder).

- Surgeweb "allow once" actions would move the selected message to the inbox (with the subject cleaned) in a way not compatible with late forwarding.
- IMAP moving a message to the inbox, would release all messages by this sender moving them to the inbox (without the subject cleaned, but in a way compatible with late forwarding), and move a duplicate of the actual selected message to the inbox with the subject cleaned.

In either case `user.cgi`, status email, and friends bounce reply based message releasing has been correctly releasing emails (without subject washing however) regardless of to the state of `g_spam_folders` or `g_spam_folders_show`.

This has been corrected twofold:

1. imap fix in surgemail (version 4.3f-10) means that the case the wrong message getting released and deleted would now display an error instead. This will start to work if you do binary only upgrade.
2. surgeweb fix (version 4.3f-12, requiring a full upgrade including surgeweb templates) which results in the behaviour below:
 - Surgeweb "not spam" / "allow" actions are always subject cleaned, all other releases (`user.cgi` / status email / imap move / friends bounce reply) may or may not be set to have subject cleaned on release using the new global setting `g_friends_release_wash`.
 - IMAP move actions ("Friends pending" -> INBOX) always move the selected message to inbox (not compatible with late forwarding), and releases any additional messages by same sender (in a way compatible with late forwarding). No duplication happens.
 - Surgeweb "not spam" in the spam folder releases the selected message and all messages by the same sender all in a way compatible with late forwarding. "allow once" is also compatible with late forwarding. In both these cases the stars (or other surgemail spam tagging) are removed and the spam folder surgeweb message list displayed is immediately refreshed. Also surgeweb displays more complete information on what it is and is not able to do in response to "spam" / "not spam" / "allow once" / "block" actions.

Recommended method of spam quarantening

This basically involves using friends silent mode instead of the spam held mode as the primary method to quarantine email identified as likely spam.

The advantages of this are:

- Reduced false positives by making use of the the Friends whitelist.
- Spam folder management using clickable links in the status email allowing management from your email client.
- In status email, can show you just the new messages received since the last status email was last sent.
- Provides a single quarantine location for the storage of all messages identified as possible spam.
- This spam quarantine location can be made available through imap and may be renamed if needed.

Why change?

The primary reason is to make use of the Friends whitelist. Traditionally surgemail has had two ways of storing spam before it gets to your inbox.

1. **"Friend pending"** folder where messages were held pending "friends bounce" confirmation that the sender was human.

2. **"Spam Held"** folder that all messages with a high spam rating were placed.

The **most effective** way to configure this has been to use a **friends bounce if over a certain spam rating**. This way all spam will be stored in the "(Friends) Pending" folder and if senders of messages that get placed in the "Pending" folder reply to the bounce the email will be immediately delivered to your inbox.

Some people try and **avoid sending out of challenge / response bounces** out of personal preference. In this case you would likely have the friends system disabled, and have messages end up in the "(Spam) Held" folder based on spam rating. The friends system intrinsically allows email based on a whitelist of valid senders. **If the friends system is in the disabled mode, mail from people you have been mailing from a long time would often end up in the "Held" folder** - just because in this case the message happened to look like spam.

In the case where you are trying to avoid sending challenge response emails **it works much better if you use the friends mode "silent"**. This way all email over a certain spam rating is placed in your "(Friends) Pending" folder (displayed in surgeweb as the "Spam" folder). This way **email from people you have mailed in the past** will always match the friends whitelist and **never get identified as likely spam**.

The second core reason is that you can use the new html formatted status emails to manage your quarantined messages directly from within your normal email client.

New format html spam status email

The old plain text status email reporting on the status of the friends pending (and spam held) folders has been replaced by an html formatted email with more information on the email messages that have been detained in the quarantined in the "Spam" [aka "(Friends) Pending"] folder.

This email provides **clickable links so spam email messages can be managed directly from your favourite email client** without needing to manually log into user.cgi / webmail / surgeweb to process messages identified as spam. This is particularly useful if you are using surgemail in a surgewall configuration for spam filtering. You will need to enable the status email on the log page in user.cgi and you need to be running a recent version of surgemail (4.0v-8+).

From: Mail Delivery Subsystem <postmaster@netwin.co.nz>
Subject: Spam Report for support-pn@netwin.co.nz (29 messages)
Date: Saturday, 24/10/2009 2:09 PM

Show: [raw](#) [html](#)

This message is an automatic status message summarizing messages detained as suspected spam and a log of important serverside email processing events. To disable or configure this message login to the [user.cgi](#) interface and click on the "Log" link.

29 new suspected spam (Total messages: 499)1 day(s) to Friday, 23-Oct-2009 20:09:06 CST

Action links	Score	Message details	Size	Expiry
Deliver to INBOX View Delete Block	13	Subject: Greetings From London & Business Proposal From: workjim@virgilio.it (Jim McConville) To: workjim@virgilio.it Reply-To: tkbnjim@yahoo.co.jp	1 KB	14 days
Deliver to INBOX View Delete Block	13	Subject: Greetings From London & Business Proposal From: workjim@virgilio.it (Jim McConville) To: workjim@virgilio.it Reply-To: tkbnjim@yahoo.co.jp	1 KB	14 days
Deliver to INBOX View Delete Block	30	Subject: Enjoy a new car, no matter your credit From: wyome.vto1638@freeautumndirect.com (AutoLoans) To: <support-webmail@netwinsite.com> Return-Path: bobxf@freeautumndirect.com	1 KB	14 days
Deliver to INBOX View Delete Block	20	Subject: Get a diploma for a better job. From: dichotomiesld@800democrats.com (Joni Salinas) To: <support-webmail@netwinsite.com>	1 KB	14 days

The new formatted html email by default allows you to take the following actions on messages using a single click:

- Deliver messages to your inbox, training the from address in your friends list to ensure future messages from this user are always delivered without being marked as spam
- View the message safely without delivery to your inbox
- Delete the message
- Permanently block all email from this sender
- There are also options to enable spam reporting and Spam folder purging from this page

Another feature of the html status email is that it displays all the relevant addressing headers if they are different from the From header to help identify mail with faked headers. In particular "Reply-To" and "Return-Path".

This status email may be customised and the basis of this email is the `surgemail/status_html.eml` file. The status emails can also be sent at a particular time of the day using the `g_spam_status_hour` setting.

Surgemail will start globally using the html status email by default. Individual accounts can be configured to send the old style plain text or new format html status email using the `user.cgi` log page. In addition the global default to start usign the html status email can be disabled using the setting:

```
g_friends_old_status_email "true"
```

Updating to new Default spam handling settings

To configure the new global default behaviour all you need to configure is `g_friends_default_mode`. This can be configured several different ways:

1. Behave like spam held, by placing spam in the Spam folder:

```
g_friends_default_mode "silent"
```

2. Use friends whitelist, and deliver spam to the inbox:

```
g_friends_default_mode "list"
```

3. Use challenge response based on spam rating by default:

```
g_friends_default_mode "smite"
```

This will enable this friends for newly created accounts. For new surgemail installations `g_friends_default_mode` silent is now the default friends mode.

In addition to this, you will need to make sure you have no global defaults or domain defaults defining spam hold settings or disabling Friends whitelist. You will be warned during the tellmail conversion below if you have any of these.

Plus optionally if you are using surgeweb convert the [relevant surgeweb settings](#).

Lastly there is one further setting (version 4.3b-2+) you may want to use to consolidate the changes:

```
g_spam_hold_hide "true"
```

If enabled this:

- Hides the Spam hold and vanish settings for end users (Spam reject is still displayed), and displays a warning in the spam settings to use the Friends based quarantining settings instead. Spam hold and vanish settings are still displayed for server admin.
- In filtering and exceptions renames Request to Quarantine and treats spam hold exactly the same as the Quarantine setting (ie the option that was previously named Request). Even though it was previously named request it only sent a challenge email if one of the friends modes was configured that sends challenge emails, so quarantine is more appropriate.
- Some rewording in the Friends settings

Changing spam settings for existing users

Changing the defaults above does not affect accounts where changes have been made manually by users at the user.cgi level.

A tellmail command has been implemented (surgeemail version 4.2g-33+) to aid the conversion process by allowing the admin to convert all users at once. The syntax is:

```
tellmail held2pend (global|mydomain.com|email@mydomain) [apply] [nocheck]
```

Where:

- apply - required to actually make changes. Without this it simply tells you what it would convert.
- nocheck - disables the checks for spam_held being enabled by default or friends being disabled by default. Using this flag makes testing the held2pend functionality on a single account easier, but normally before use on multiple accounts these checks should get passed rather than ignored using 'nocheck'.

eg.

```
>~ tellmail held2pend global
```

```
Converting accounts using spam_held to using friends_silent.
WARNING: Running test only, rerun with APPLY parameter to actually convert accounts.
```

```
Processing domain (mydomain.com):
user1@mydomain.com: 283 msg; friend mode=disabled->silent score=5->10 add_auto(enabled); {wrap}
                        spam H/R/V= 10/0/0 -> 0/0/0
user2@mydomain.com: 15 msg; friend mode=list score=5->8; spam H/R/V= 8/0/0 -> 0/0/0
user3@mydomain.com: 94 msg; friend mode=smite score=7; spam H/R/V= 15/0/0 -> 0/0/0
```

```
Processing complete.
```

```
>~
```

In this case:

- User1 had friends disabled and was using spamheld on a scoring of 10 or higher. This has been converted to using friends silent mode, at rating of 10+, with addresses of outbound mail getting added to the whitelist.
- User2 was already using friends whitelisting (not using the score value at all though), but now the spam gets stored in the friends folder rather than the held folder at a rating of 8+.
- User3 was already using friends challenges at a rating of 7-15, with 15+ getting stored in spam held. Now any mail with a rating of 7+ gets sent a challenge and is stored in the friends pending folder.

after which you will need to run:

```
>~ tellmail held2pend global apply
```

What the conversion actually does is:

1. Switch friends disabled or list modes to list or silent dependent on the spam_held rating. Other user configured friends modes are retained.
2. Use the lowest of the relevant processing ratings (held/vanish/bounce or friends smite rating) and use this as the friends smite rating. Possibly keeping the spam_reject setting if relevant.
3. Enable friends whitelist addition based on outbound authenticated smtp.

I think what it does makes sense, but do check up on this. All the settings that it changes (or are relevant) should

be listed in the tellmail command output. Feel free to contact surgemail-support@netwinsite.com to discuss any questions.

Two additional relevant tellmail commands:

held2pend_email Set spam held email frequency for all accounts. days="days between emails" with special values 0="disabled" and -1="server default"

```
tellmail held2pend_email (global|mydomain.com|email@mydomain) days=n [apply]
```

pending_release Release mail (without witelisting) mail in users' friend pending folder. nofriend="only apply to accounts without custom friends settings" all="apply to all accounts", cutoff_rating="smite rating below which to release messages (optional)"

```
pending_release (global|mydomain.com|email@mydomain) (nofriend|all) [cutoff_rating] [apply]
```

Word of warning

Surgeemail will automatically switch to using the html status email in versions (4.0v+), and will only send this if there are messages to report on. This was recently identified as having the nasty and unintended side effect in surgemail versions (approx 4.0v to 4.2g-26) that for accounts using the status email to report on messages only in the spam held folder (and not using friends) the **status email will appear to stop getting sent**.

This is resolved in version 4.2g-27+. This version also introduces per account control over whether the new format html or old format plaintext status email gets sent.

SurgeWeb Advanced Customisation

It is strongly recommended not to change anything in the surgeweb/tpl directory. This page will show you how you can do advanced customisation without needing to modify the tpl directory.

You can enable the extension css and javascript instead. These will allow you to override any surgeweb core CSS declarations or replace any native surgeweb javascript functions. You can customise any core surgeweb javascript function. Either by replacing it or more likely "hook" them by providing your own intermediate level of functionality which then calls the core surgeweb code in order to do the core interface actions.

"I want to change logos, colours and 'look and feel' of the interface."

You probably should not be looking at the advanced customisation techniques but be looking at the [standard customisation page](#). The admin interface surgeweb customisation page allows you to change much of the look and feel without needing to consider html, css, and javascript. Also you can use skins to define a set of styles to be applied to multiple domains

"Ok, but I really, really want to change more than that."

Below are the recommended ways of making these changes to keep your life easy during surgemail upgrades. If there is something you want to change that you cannot, please contact surgemail-support@netwinsite.com with a description of what you would like to be able to change and why.

If netwin staff are troubleshooting your system for surgeweb problems and advanced customisation has been applied, one of the first tests will be to test with customisation disabled. This will preferably be done using the surgemail.ini setting surgeweb_custom for this purpose, or if necessary by copying the original files back into the surgeweb/tpl directory.

Here is a list of things other admins have wanted to do. If you do them as described below they will remain intact on surgemail upgrades:

I don't want users to adjust their advanced preferences

The advanced options and customisation pane in the preferences will not be displayed if you add this to any config_*.dat file:

```
lock_options_basic true
```

I want to hide advanced "disable smarts or restore older behaviour" preferences

Add this to any config_*.dat file:

```
lock_options_advanced true
```

and the following to extend.css

```
.nosmart{
    display:none;
}
```

Users should not be able to modify their reply name

To any config_*.dat file add:

```
lock_name_auth true
```

Users should be able to modify their reply email address

By default users are not allowed to spoof their from header reply email addres. To allow users to set, add the

following setting to the users' _user.dat file or to any config_*.dat file:

```
allowed_from *
```

or

```
allowed_from *@mydomain.com
```

I always want the quota displayed

A quota warning will be displayed if you go over 80%, this is normally hidden to keep the interface clean. This can be always shown by adding this to any config_*.dat file:

```
always_show_quota true
```

Display filestore as well as / instead of photos button in application menu

In extend.css add the relevant style definitions below:

```
#app_filestore{
    display:block;
}
#app_photos{
    display:none;
}
```

Remove Calendar / Photos / Blogs Application menu buttons

Calendar and photos can be hidden by disabling surgeplus using the surgemail.ini setting:

```
g_disable_surgeplus "TRUE"
```

or by using user_access settings

```
!surgeplus
```

Blogs can be hidden using the user_access setting

```
!blogs
```

Remove the remember me login box on the front login page

In extend.css add:

```
#remember_me{
    display:none;
}
```

I want the help links to point at my own website

Setup your own pages of the same name and point all the surgeweb help links at that using the following setting in any config_*.dat file:

```
help_url http://mywebsite.com
```

I want a shared addressbook to be displayed

A variety of options are available for shared addressbooks. Either Global / Domain surgeweb addressbooks or LDAP or Authentication database based addressbooks. These are configured using the abook page in the surgemail admin interface:

```
abook name="Global" read="!*" write="!*" type=""
abook name="Domain" read="!*" write="!*" type=""
abook name="LdapAbook" read="*" write="" type="ldap"
abook name="AuthentAbook" read="*" write="" type="auth"
abook name="Surgeweb" read="*" write="" type=""
```

I do not want any shared addressbooks to be displayed

Make sure the users do not have access to any addressbooks you have defined in surgemail.ini. To remove the Global and Domain addressbooks set:

```
abook name="Global" read="!*" write="!*" type=""
abook name="Domain" read="!*" write="!*" type=""
```

I do not want to offer labels in surgeweb

Labels functionality can be disabled by adding this to any config_*.dat file:

```
labels_disable true
```

I do not want to display any snippets

Message body snippets functionality can be globally / per domain disabled by adding this to any config_*.dat file:

```
snippets_disable true
```

I want different / no default groups in the contacts management

The default groups can be defined using this setting in any config_*.dat file:

```
contact_groups Friends,Family,Colleagues
```

or

```
contact_groups Foo,Bar
```

or

```
contact_groups
```

I want to remove the raw user.cgi settings

The raw user.cgi settings on the "Options - Preferences - Filtering and Spam Control" page already get hidden if g_user_access / user_access settings are used to disable spam handling. To always hide, add this to extend.css:

```
.raw_user_cgi{
    display:none;
}
```

Remove surgeweb spam handling disabled warning

The warning "some SurgeWeb spam control options disabled by administrator" is displayed on the "Options - Preferences - Filtering and Spam Control" page if the recommended surgemail spam handling techniques have been disabled. If for example you have your own spam handlign technniques in place, this wannring can be removed by adding this to extend.css:

```
#opt_fs_disabled_warn{
    display:none;
}
```

Remove processing actions on Filtering and Spam control Page

This functionality can be enabled / disabled using g_user_access / user_access / the authentication databse user_access fields and they will be removed from display in surgeweb:

```
relevant fields: fwd, fdwonly, exceptions, friends, spam
```

Remove the processing actions link

To remove the "Check the delivery log for a record of processing actions" link on the "Options - Preferences - Filtering and Spam Control" page, add this to extend.css:

```
#opt_fs_log{
    display:none;
}
```

Remove the display of running surgemail version

To remove the "Options - Preferences - Advanced" page current surgemail version, add this to extend.css:

```
#opt_adv_version{
    display:none;
}
```

Modify the banner image in mobile interface

```
config_domain.dat add:
    mobile_custom_banner true
and you place your modified logo image in
    surgeweb/custom/banner_mobile.jpg
or
    surgeweb/custom/your_domain/banner_mobile.jpg
```

Arbitrarily modify mobile template

Enable mobile customisation using config_*.dat setting of: (Surgemail 5.3c-24+)

```
mobile_custom true
```

Then create _mobile_custom.htm file in surgeweb/custom (and optionally in surgeweb/custom/domain.com). This allows you to arbitrarily modify the icon url or completely override and replace the content on individual top level template pages.
eg.

```
# modify the banner image url as we see fit if needed, here are two examples
#|define((mobile_banner),concat(/whatever?domain=),domain_ex,(&img=banner_mobile.jpg)))||
#|define((mobile_banner),(http://netwinsite.com/img2/logo_med_onwhite.png))||

# Example of overriding the whole template output of any of the top level pages in
# the surgeweb/tpl/mobile directory
||if(equal(page,(sw.htm)))||
    Yes!! Own template output :-)

#      Should base page content on the original file sw.htm in surgeweb/tpl/mobile directory
#      Can even include a template from the original surgeweb/tpl/mobile directory if we want
||include||../tpl/mobile/_msgs.htm||

    </body>
    </html>
    ||template_exit()||
||endif||
```

Note: This mechanism for template customisation without making changes to the tpl directory can be used on the html interface as well using the settings of "html_custom true" and "_html_custom.htm" file.

I want the change password button to go to my own interface

Enable extend.js and enable this code:

```
# Hook the usercgi dialog function to do some of our own handling instead
if(true){
    hook_user_cgi_dlg();
}
# Store original function and replace with our own one. In this case do our own action if it was the password
button
var user_cgi_dlg_orig;
function hook_user_cgi_dlg()
{
    user_cgi_dlg_orig=user_cgi_dlg;
    user_cgi_dlg=function(type){
        if (type=='password'){
            open_popup2('https://my.server.name/public/changepass.aspx');
            return;
        }
    }
}
```

```
        user_cgi_dlg_orig(type);
    }
}
```

I want to hide the change password button

There are two ways. Either set !pass in the accounts user_access settings in the authentication database, or add this to extend.css:

```
#opt_gen_change_password{
    display:none;
}
```

I want my own link(s) to the right of the username at the top of the page

Add this to extend.js.

```
var node=dge('link_bar').childNodes[2];
var frag=document.createElement('span');
frag.innerHTML=' <a href="http://netwinsite.com">MyLink</a> | ' ;
node.parentNode.insertBefore(frag,node);
```

I want to customise the folder icon

Add an appropriate image to your custom directory and then add a rule with appropriate offsets to extend.css. (currently the default file you may want to base your icons on is `surgemail\surgeweb\tp\shared\img\vbtn_icons.gif`)

```
.folder_icon{
    background: url(vbtn_icons.gif?|vz|) no-repeat 0 -80px;
}
```

I want to change the folder list text colors

This may yet be changed to simplify it but adding these to custom.css will allow you to customise this currently.

```
# Base folder name colours
ul.ftree{
    color:blue;
}

# Colors associated with folder list selection handling
.ftree .selected, .ftree .hover_js, .ftree .hover_js a, .ltree .selected, .ltree .hover_js {
    color:blue;
}
ul.ftree .selected a {
    color:blue;
}
#folders .part_cursor:hover, #folders .part_cursor:hover a {
    color:blue;
}

# Labels and Searches text color
.green_text{
    color:blue;
}
```

I want to hide the report via email button

To custom.css add: (Surgemail 5.0h-10+)

```
.fault_report_button{
    display:none;
}
```

I want right column on login page too

To config_*.dat add: (Surgemail 5.2d-3+)

```
rcol_login Some text
```

or iframe based:

```
rcol_login iframe
rcol_login_url http://netwinsite.com
```

and in extend.css customise rcol_login_outer and rcol_login_iframe for positioning etc as needed.

My security audit requires login page autocompletion to be disabled

To config_*.dat add: (Surgemail 5.3c-26+)

```
disable_login_autocomplete true
```

I would like to add my own hints to the surgeweb hints

The surgeweb hints system (Surgemail 5.3i-67+) is easily customisable with your own hints. To do so you will need to do two things:

1) Define the actual hint strings in one of your lang.dat files (eg custom/lang.dat)

```
st_hint_custom_01 Hint: The My Company mailserver documentation can be found <a
href="http://mycompany.none/page.htm" target="popup">here</a> and is maintained by John Doe
st_hint_custom_02 Hint: This rather important thing our My Company users should know about...
```

2) Define the strings id numbers you want to get shown using a setting in config_*.dat

```
custom_hints_show 01,02
```

Note: a maximum of 32 custom hints are available and the numbers must be two digits (ie 1-9 are zero prefixed)

I would like to hide some / all of the hints

System hints can be individually disabled using the following config_*.dat setting, where the numbers refer to the hint numbers as defined in surgeweb/tpl/lang.dat:

```
system_hints_disable 01,02,05
```

Disable the hints system completely using (surgemail 6.0b-4+) [and 6.0a3-3(patchd)]:

```
all_hints_disable true
```

I would like to customise the idle autologout

Idle autologout can be enforced for all users of a domain / globally using config_*.dat setting of: (surgemail 6.0b-4+)

```
autologout_enforced true
```

Alternatively assign the default duration (in minutes) and default state for new users [on|off] using these config_*.dat settings:

```
autologout_duration 30
pref_autologout_type on
```

I would like to disable Extras tab

The extras tab in preferences (introduced surgemail 6.0b-5+) can be disabled in part or completely, at the surgeweb level using comma separated entries as follows [all|legal|notify|lists|import|alias] eg

```
extras_disable all
```

or

```
extras_disable legal,alias
```

None: Extras tab also only shows the relevant entries when allowed to as per user_access authentication database permissions

What else can I do with this?

Here is an example of a very different application menu. The menu has been moved to the topleft instead of bottom left and is image based. To try this, enable extend.css and extend.js, then uncomment this is code in extend.js:

```
if(true){
    move_panel_to_top('app_menu')
    alternate_menu();
}
```

Or alternatively here the creation of panels on the left say with your own content in it. Here are several examples from extend.js:

```
#      New fancy panel at the bottom with title and body text. Again can have anything in it
new_panel_fancy('My own panel header', 'Fancy panel body with anything in it','bottom');

#      New fancy panel at the bottom with live maps as an example
var content='<iframe id="a_map" src="http://wikimapia.org/s/#lat=-40.7805414&lon=173.8476563&z=4&l=0&m=a&v=1"
width="'+lc_width()+'" height="200" frameborder="0"
style="background:|lpanel_color|"></iframe>';
new_panel_fancy('Wikimapia!! <i><a href="http://wikimapia.com"
target="_popup">link</a></i>',content,'bottom',true);

#      New fancy panel at the bottom with youtube content as an example
var content='<object width="'+lc_width()+'" height="200"><param name="movie"
value="http://www.youtube.com/v/Pfs4Rd5f_IQ&hl=en&fs=1"></param>'+
'<param name="allowFullScreen" value="true"></param>'+
'<embed src="http://www.youtube.com/v/Pfs4Rd5f_IQ&hl=en&fs=1"
type="application/x-shockwave-flash" allowfullscreen="true"'+
' width="'+lc_width()+'" height="200"></embed></object>';
new_panel_fancy('Youtube!!!',content,'bottom',true);
```

Emergency disabling behaviour for performance reasons

Some of surgeweb's more cpu and disk io intensive smarts can be globally disabled. Warning this should only be done for troubleshooting reasons generally - and probably only as a last resort if you have investigated all other sources diskio and cpu bottlenecks.

```
# Globally disable inbox caching. Anyone saving preferences with this enabled will set
# their account's inbox caching to disabled in their own preferences. This will remain
# disabled unless the user re-enables.
global_nocache true

# Globally disable new mail checks. Anyone saving preferences with this enabled will set
# their account's inbox caching to disabled in their own preferences. This will remain
# disabled and the user is not able to re-enable, so requires editing of _user.dat to
# re-enable new mail checks.
global_nocheck true
```

Anyway, as I said at the start of the page. Any customisation done this way will remain intact on surgemail upgrades. Please contact us on surgemail-support@netwinsite.com if you have something you feel you need to customise and currently cannot, explaining exactly what you want to do and why.

Marijn

SurgeWeb Banner Advertising

SurgeWeb is designed to make it easy to add banner advertising to your page. Basics of google AdSense based advertising are outlined below. If you have need for other forms of banner advertising please contact surgemail-support@netwinsite.com with the details and an example of how you would normally integrate this, and we will offer suggestions on implementing this in surgeWeb.

Google AdSense advertising

SurgeWeb google AdSense advertising can be enabled in a right column in the interface by setting the "right_column" content setting and then customising "ad1_html" to contain your google AdSense string "all on one line".

eg

```
right_column image_ad1
ad1_html <script type="text/javascript"> google_ad_... /show_ads.js"></script>
```

Getting the ads to "rotate"

First a warning: SurgeWeb is an Ajax web application and only has a single page refresh at the point of session login. Google AdSense is not very well setup to work with Ajax applications as the googles policies basically seem to state one display of adverts is allowed per impression. Where an impression is a full page refresh thus one refresh per surgeWeb session. This seems unreasonable and all ajax application fall under the same category - including gmail.

SurgeWeb has settings that will allow the google ads to be updated on display of new folder contents, display of messages for viewing or for composing / editing. These are pretty much the points at which a full page refresh would occur if surgeWeb had not been implemented using ajax. I am no legal expert and I don't know whether you will get in trouble with Google if you use these settings, but I would have said there is a reasonable case for "fair use".

Anyway, as now setup in surgeWeb there are two settings that can be configured on the surgeWeb customisation page in surgemail:

- ad1_updates: possible values 'rotate', 'target_simple', or 'target_subject'
- ad1_keywords: string of comma separated words eg 'auckland,amsterdam,new york'

Set to '**rotate**', the ad banner will rotate on the interface actions described above but no targeting of the ads is attempted. Dependent on your website this will provide different ads but in my experience leans to many "electrical surge protection" ads as "surgeWeb" is part of the url. That also might just be my test server has not been crawled by google, but google should never get much by trying to crawl surgeWeb pages. Maybe the higher level pages??

Set to '**target_simple**' this will step through each of the ad1_keywords words when updating the ad banner, and use it as part of the "site url". This should provide a basic level of targeting, although there seems a fair bit of black magic in what google displays.

Set to '**target_subject**' will try and use the message subject to target the message for clicks opening a message. Other clicks like folder clicks will default to the ad1_keywords defined words, and without that default to whatever google serves.

Browser Bugs affecting SurgeWeb

Google Chrome 3.0 drag and drop issue

SurgeWeb drag and drop clicks in Google Chrome 3.0, triggers some form of browser bug that results in other events getting sent to the wrong parts of the interface and results in very strange behaviour.

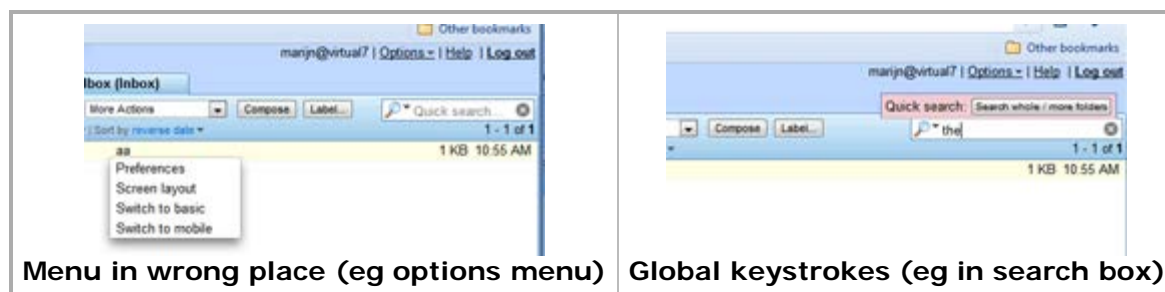
Affected browsers: Google Chrome 3.0

Does not affect: Google Chrome 4.0, 2.0 and 1.0, Internet Explorer, Firefox and Safari.

Issue description

Examples of odd behaviour are:

1. Menus appearing in the wrong place.
2. The global keyboard actions getting triggered when entering text in edit fields such as the search box or password relogin box. Most notably "n=compose new message", "r=reply to selected message", "delete/backspace=delete selected message", and the other keyboard shortcuts defined as documented in the Options-Preferences-Customise page.



Menu in wrong place (eg options menu)

Global keystrokes (eg in search box)

This will happen after any actual drag and drop operation (even just clicking) of messages in the message list or contacts in the contacts list. This will continue to happen until one relogs in to surgeWeb.

Mitigation

The work arounds are:

- Disable drag and drop on login when warning is shown
- Do not use these actions on Chrome 3
- Use another browser including Chrome 4

SurgeWeb

Fast Ajax web email

Better than a desktop mail client!

SurgeWeb is our brand new web email interface that uses Web2 / Ajax techniques to provide the responsiveness, performance and more features than most desktop email clients.

Modern web application features

Modern features such as address autocompletion, message organisation via drag and drop, right click context menus throughout the interface, full cross browser html editor, optional automatic image downsizing when sending large images, support for sending either html email or format flowed plain text email, superbly suited to higher latency network connections.

Clean and elegant interface

The interface is clean and elegant with many actions performed unobtrusively in the background. Background actions include: new mail checks with audio notification of new messages, automatic background saving of drafts, and background uploading of attachments as you continue to edit your email, and actual message sending.

Excellent Filing and Searching of mail

Includes collapsible nested folders, unlimited arbitrary message labels, super fast indexed searching of subject / to / from headers. Search based smarts such as single click "all mail with this person" or "all messages in this thread". Supports rich search syntax, and search results can be saved for later one click access.

Multilanguage support

The surgeweb interface is easy to customise for the international language of your choice. By default surgeweb comes with partial sample translations of 26 non English languages. Separate support for English UK and English US.

Tabbed and popup interface design

Supports tabbed mode editing of multiple messages in a single browser window (with a button to popout individual messages into their own window), or automatically edit all messages in their own popout window.

Easy to customise

Surgeweb uses techniques to layer customisation over the top of the base interface using variety of settings, CSS, javascript and skins that can all be applied at the global, domain or user group level. This also provides for easy upgrades of customised surgeweb installations without needing to "reapply" a lot of template customisations.

Shared addressbooks

Surgeweb supports shared addressbooks at the global, domain or user group level. Also supports external LDAP based or surgemail authentication database based shared addressbooks.

Fast and efficient

Many times faster and more efficient than the old NetWin Webmail, allowing you to host more web based mailboxes on the same hardware.

Will work with your existing IMAP mailserver

Surgeweb is designed to be used with surgemail but will also work with your existing IMAP mail server and can provide a high performance modern webmail client in minutes.



previous **CUSTOMER COMMENTS:** next

What I like most in SurgeWeb... mostly that it "feels" modern. The functionality is crisp, clean and intuitive.

Other features that are great: The "multi-tasking" aspect, the address auto-complete, the real time spell check, the fast speed, the "not losing work" if logged out, and yes, the way SurgeWeb is pushing us towards a more basic approach to spam (although that one took me a while to appreciate, I really do now)... the drag and drop, the customization aspects of the GUI.

I know lots more will come to me as I use it more :-)

-- Byron Knapp - Kargo.net

If you have any questions we can answer, please do not hesitate to Email us at: surgeweb-support@netwinsite.com

Satisfaction Guaranteed! We stand behind our software, if you are not satisfied with the product, performance or customer support we will refund your purchase or work to resolve whatever issues you raise.



Related Pages
[Surgeweb manual](#)

YesImOnline Beta Testing!

Make your daily EMail efficient!

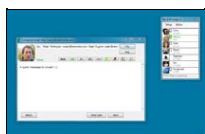
Our basic premise is that there are two types of email:

1. Regular boring email to which you must respond a few times a day at most.
2. Urgent email from co-workers or the wife which must be seen and answered within minutes.

YesImOnline is a simple email client that works with your existing email client (or stand alone) to help you respond rapidly to important email. It also lets you see when your important contacts are online. (Strangely one emails differently knowing this, for example I don't ask someone a question at 10:00 p.m. unless I happen to know they are working late)

In brief, it makes your email more 'immediate' almost like instant messaging, without the problems inherent in instant messaging (namely that IM wastes a lot of time and interrupts your work flow)

YesImOnline is a bit like a virtual office. It lets you feel the presence of your colleagues and accelerates important work by decreasing the round trip time communicating with those key people.



Windows

[Update History](#)



OSX

[Update History](#)
[More Info](#)



iPhone

Purchase in AppStore
(Free version available soon)

[Update History](#)



iPad

Purchase in AppStore
(Free version available soon)

[Update History](#)

Please note this product is still in Beta Testing, use entirely at your own risk :-)

Features Include

- Fast and simple messaging
- Instant notification of new messages (push notification on iOS), ideal for use with SurgeWeb
- Visual and Audio notification on new email from selected people
- Reminder messages (when you are busy you can make an email 'vanish' and come back later)
- Simple fast 'one click' processing of email.
- Online status notification for your favourites/colleagues. (so you know when they are available)
- Works stand alone, or with your existing email client.
- Works on any IMAP email server.
- Completely FREE :-)

Related Pages

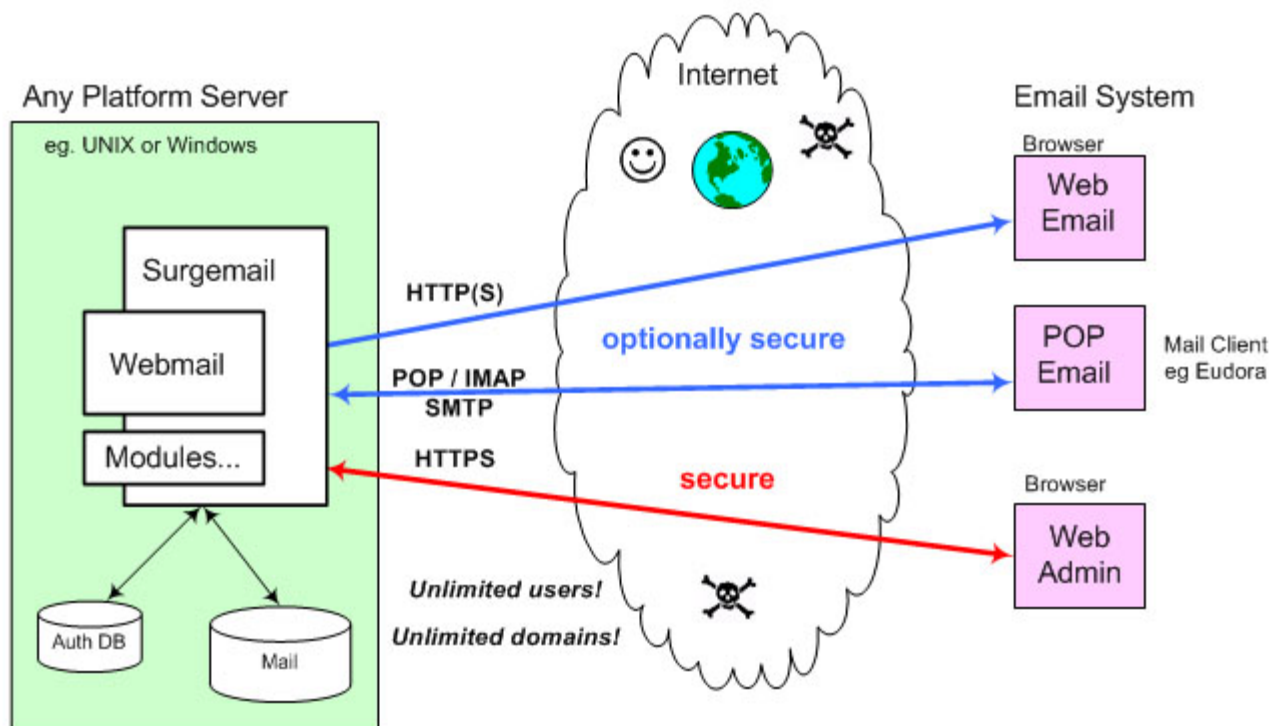
[YesImOnline manual introduction](#)

What is SurgeMail?

SurgeMail is a fully featured enterprise class mail server with integrated web based email, web based account and server administration. You can even configure it out of the box to automate user sign up (great for running free/low cost public mail services like hotmail)

The web based email system builds on NetWin's well known [SurgeWeb WebMail](#) system with extensive customization options using templates. Combine this with SurgeMail's inbuilt web server and you have an all in one solution that does not require extensive work to get components working together.

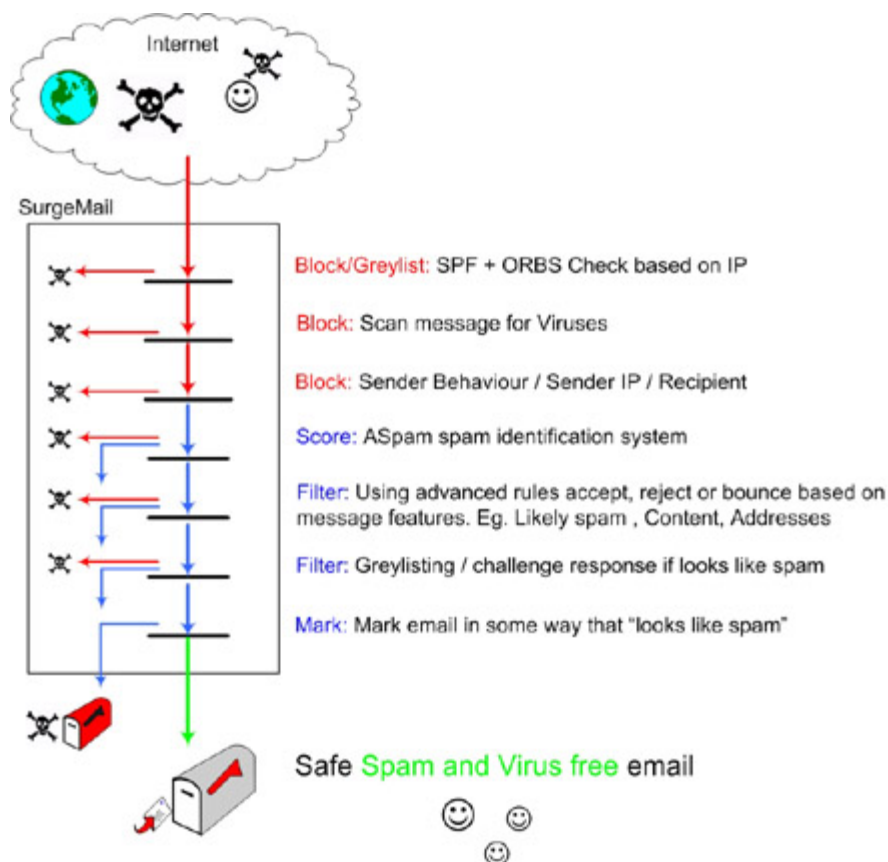
SurgeMail has many features for high reliability systems such as in-built server mirroring and support for clustered and proxy configurations, NFS based mail storage etc.



Next: [Spam and virus free email](#)

SurgeMail spam and virus prevention

This email server offers advanced features to identify undesirable spam email, block virus infected mail and prevent abuse of your mail server by spammers. Naturally these features can be individually configured. In addition you can choose between several virus scanner options (Avast, F-Prot or any command line scanner)



Open Relay database check + SPF checks

SurgeMail has integrated support for a variety of sender identification checks. This includes:

- Integrated [Open Relay](#) check connecting to any external ORB database to ensure the sender is not blacklisted for sending spam email.
- Integrated [SPF](#) checks to verify the sender is actually who they say they are.

If a message fails one of these checks the message can be bounced or rejected immediately or this result can be used to add to the ASpam spam detect score to reduce false positives.

Sender behaviour limitation

SurgeMail email server has dozens of configuration [options](#) to directly block or tarpit users or servers identified as abusing the mail server. These include banning by sender IP address, recipient mail address, from mail address, limits on number of recipients per email, maximum bad addresses in a row, email per sender IP address and maximum messages to a single mail account.

Virus scanner integration

SurgeMail email server will integrate with any external command line **virus scanner** that has the option to delete mail if it contains a virus. eg: This allows you to use for example Command, Sophos or Norton virus scanners. See [here](#) for details.

For Windows we recommend "**Avast! for SurgeMail**" based on [ALWIL Software](#) antivirus technology which is fully integrated. See [installation and configuration](#).

F-Prot is also supported for UNIX and Windows, [see here for details](#).

ASpam anti-spam system

SurgeMail has built in support for [Aspam](#). This is a message "spamminess" scoring system based on the sum of the following:

- Customisable rule database maintained by netwin staff - This is approx 60% accurate on common spam.
- Auto training database of recent messages that "look like spam" based on poly and multi symbol statistical word matching.

Approx 90% effective if no local training is done, approx 99% effective if local training is done.

- Auto training database of recent messages that "look like spam" based on message parameters such as URL content. Approx 40% effective if no local training is done, approx 99.5 effective on trained data.
- Catcher addresses that should never receive genuine mail, and if mail is received on these addresses it is known a spammer.
- Optional modification of scoring based on ORBS and SPF checks.

The auto training databases consist of a base set of rules maintained at netwinsite.com combined with local training based on messages submitted by the users of your system as uncaught spam or as a false positive.

Based on this "SpamDetect score" messages can be filtered at a serverwide level or at a per user level allowing individual users to fully customise their filtering setting up a totally customised "personal antispam policy" based on their chosen level of spam 'tolerance'.

Advanced mail rules

Using elaborate [rules](#) customised policies can be setup for mail forwarding, archiving and filtering. Filtering will typically take some form of action on messages identified as spam by [SmiteSpam](#) or external spam identification filters. An alternative use for filtering is to limit mail based on content. This can be internal to surgmail using [mfilter rules](#) or externally using your own filter application using [g_filter_pipe](#).

eg: This could allow you to setup policies that

- Mail identified as almost certainly spam is either dropped or bounced
- Mail with undesirable subject matter in body or subject line is bounced

Friends only system

The [friends](#) only system is a challenge response system allowing users to opt to **receive messages only from friends**. Non friends are automatically questioned to determine if they are human. All mail from non friends is held pending on the server until the user has decided what to do with it. Status reports are sent to the user on a regular basis to provide information on the Friends system and any mail pending delivery.

See our [brief guide on stopping spam](#) for existing and new users.

Next: [Extended SPF](#)

Web based administration

The new web based administration gives you the flexibility to do all server administration tasks using the integrated web based administration interface. Administration features fall in three categories

User self administration Which allows the user to update their account information and (if enabled) create and delete accounts their own accounts.

Domain administration: Domain administrators have the facilities to manage the accounts of their users.

Server administration: System level administration can also be done using the fully featured web interface. This includes restarting SurgeMail, stopping SurgeMail and if ever necessary starting the server when it is not running.

Next: [Unique Mirror system](#)

Mirror (Replicate)

The SurgeMail 'Mirror/Replication' system allows you to link two systems together and read or deliver email to either system, both systems will continually 'match' each other. This can be used in several ways:

- Keeping a live backup system for 'hot swapping'
- Keeping a live backup system with router based instant failover
- Moving a system from one geographic location to another with no downtime.

[See manual technical information on 'mirror' settings](#)

Next: [Full TLS/SSL support for all protocols](#)

SSL secure email support for all protocols

SSL is fully supported secure email on all protocols to ensure sensitive email messages or passwords cannot be read by hackers or users who may have access to the communication channel between you and the mail server. If you are running a mail server that doesn't support this feature then essentially anyone with access to your network can steal passwords. You can specify whether to allow users to login in non secure mode or not with IP range limits too. Almost all popular email clients now support SSL/TLS.

POP: Secure to regular port using STARTTLS, secure to dedicated port.

SMTP: Secure to regular port using STARTTLS, secure to dedicated port.

IMAP: Secure to regular port using STARTTLS, secure to dedicated port.

HTTPS: All web based administration tasks can be done either using secure HTTPS or standard HTTP.

Mirroring / replication: The in-built server mirroring feature mirrors the server over a secure link.

Next: [Interface to existing Authentication databases](#)

SurgeWall

SurgeMail filtering for your existing mail server

SurgeWall works with **your existing mail server** so it can be installed in minutes, but instantly gives you the uniquely advanced, [Spam Blocking](#), [Virus Scanning](#) and even user configurable [filtering and friends](#) modes normally only found in our SurgeMail product.

- [SurgeWall - how to configure it](#)
- [User existence is checked at 'rcpt' stage to prevent major spam problems](#)
- [Spam and virus free email](#)
- ['Friends' mail screening system.](#)
- [Integrated mail server and web based email](#)
- [Web based administration](#) (user, domain and server)
- [Full SSL secure support for all protocols](#)
- [Runs on platform of choice](#) (Windows, most Unix versions)
- [Installs in minutes](#) and unparalleled after sales service

[Download Now!](#)

Consulting - Remote Management - Mail Service Outsourcing

We also offer a full range of consulting and management services, from installation on your own system to full managed server contracts, email consulting@netwinsite.com for a free quote or system design.

If you have any questions we can answer, please do not hesitate to Email us at: support-surgewall@netwinsite.com



Satisfaction Guaranteed! We stand behind our software, if you are not satisfied with the product, performance or customer support we will refund your purchase or work to resolve whatever issues you raise.

Easy to install

SurgeMail is easy to install. In fact, within minutes you can have a complete multidomain "hotmail" style email system setup and running.

After Sales Service

We pride ourselves in giving our customers unparalleled service. If you have questions, problems, or even need a feature added in a hurry we will bend over backwards to satisfy your needs.

Go on and give it a try for free!

[Select download](#)

Interface to existing Authentication databases

SurgeMail has its own high performance user database system built in but can easily be configured to use existing user authentication databases, or you can configure SurgeMail to access your old POP3 server and then auto create accounts locally.

Other authentic modules supported include:

- Windows NT user accounts
- Unix user accounts
- ODBC account database
- LDAP database
- MySQL account database
- Your own proprietary system

In fact using the new MultiAuth module you could configure SurgeMail to authenticate against any number of these combined. Although this is not the suggested way of doing it as it makes the central management of accounts more difficult - we recommend picking one!.

[See the manual and download a module](#)

Per account services

SurgeMail can be configured to give access to certain mail services (WebMail, POP3 server, IMAP server, or SMTP server) on a per account basis. This could for example limit certain accounts to WebMail only, or allow you to charge a premium for POP access to accounts.

Next: [Scales to any number of users](#)

Scalability

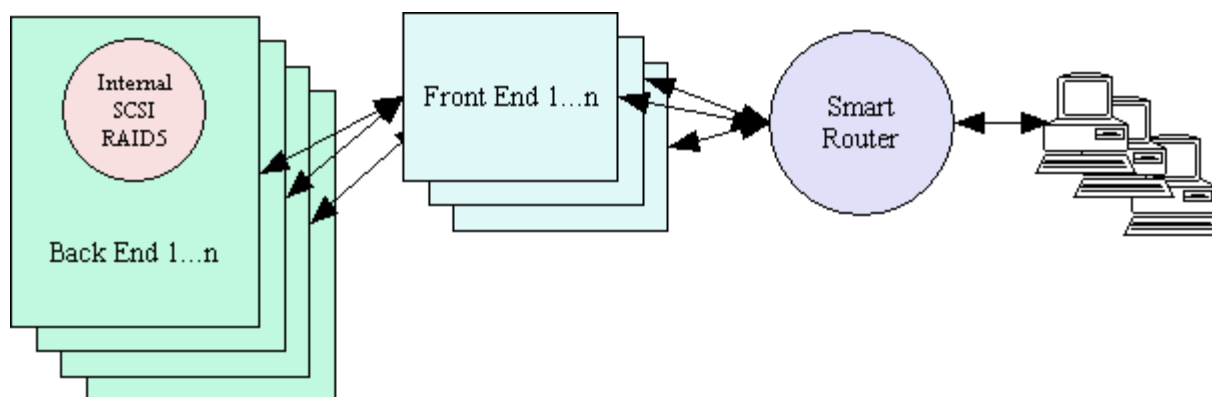
SurgeMail has been designed from scratch with performance in mind. This means you can almost literally host "an unlimited number of users and virtual domains on a single system". However we also had in mind ridiculously large systems, with 1-100 million users, in these situations there are two methods you can use.

- Proxy Servers (huge systems)
- NFS / Shared Networked drives (huge systems)
- Simple splitting up of function (medium sized systems)

Proxy Servers (huge systems)

SurgeMail allows both infinite scaling, and 3 layer security. The incoming POP/SMTP connections arrive at one of several front end 'proxy' servers (running SurgeMail in proxy mode) these servers then lookup the user in the networked user database (via LDAP or our own TCPAuth module) and along with the normal response an extra response code of 'tohost=backend.host.name' is returned, the proxy then redirects the user to the appropriate back end system.

So you might run 4 back end systems, each with 100,000 users, and 2 front end systems. To add more users you just add as many front end and back end servers as needed to cope with the load.



Each user is only on one of the back end systems, the only piece in the system that has to handle all the users is the user database, which is a relatively trivial task as the quantity of data per entry is so small. We recommend the use of NWAAuth or LDAPAuth but any of the database back end authent modules would be suitable.

[See here for technical details](#)

Note: 3 Layer Security: This model is called '3 layer security' as the front and back end systems can be separated by another fire wall. And in the case of 'WebMail' the user web interface can also be separated from the front end systems by a fire wall, hence '3 layer' :-)

To implement this system set on the proxy system the setting `g_proxy true`, and in the authent module add the 'tohost=xxx' field. For existing user accounts you can define `g_proxy_default host.name` so that user records with no 'tohost' entry are correctly sent to the existing back end system. In this way a non proxy based system can be instantly turned into a proxy based system.

NFS/ Shared Drives, Clustering Support

In this mode you simply define your main drop path for a domain as a networked drive, and setup multiple systems using that same drop path. As SurgeMail uses a 'maildir' directory format there are almost no locking issues even with bad NFS implementations (and most NFS implementations are a little dodgy :-).

Next: [Runs on platform of choice](#)

SurgeBlog - A High performance standalone blog server.

Consultant offline

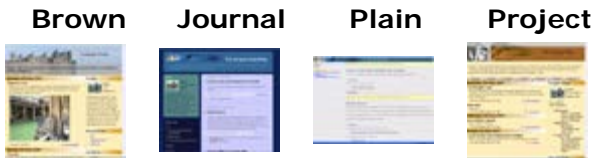
Features:

- Product Links
- [Download](#)
- [Email Support](#)
- [Manual](#)
- [Purchase](#)

- Very high performance, full featured blog server, integrated with email accounts.
- How to install and run your own Blog Server in 5 minutes. [See Setup Blog Server](#)
- User can create one or more BLOGs within WebEmail interface or from account self management interface.
- Use as companion to exchange or any other email server
- A range of blog layout templates are provided and users can switch templates at any stage.
- Posts can contain images and other attachments.
- Post directly to the blog using an email address with PIN, including images and other attachments.
- Web post interface, management interface and comments interface.
- The user can tailor the templates at various levels from simple to advanced.
- Team system to allow teams to share a blog at various levels; posting, moderation and full management.
- Extensive blog interface allows straight forward moderation of posts and comments.
- The user can define various settings and add 'team members' who can also post or configure the blog
- The blog can be regenerated with a new template set at any time.
- Automatic archiving, visit counting, and comment structure.
- No template parsing on view provides high performance, low impact viewing.
- Optional thumbnails image rotation and auto image resize provides simple posting of multiple images.
- Replace your current server with SurgeBlog/SurgeMail
- Highly competative pricing US\$570 for 1000 user license. [\(see surgemail prices for details\)](#)

For technical details and help see the [Blog Admin Manual](#) and the [Blog User Manual](#).

Some of templates available, click for larger view.



Sample Screen Shot -- Blog Settings window for configuring a Blog

List My BlogsViewNew PostList/Edit PostsSettings for lookmire

Blog lookmireSave SettingsEdit IntroEdit ProfileAdvancedHelp

TitleLookmire Family-

Urlhttp://ralphslaptop/blogs/lookmire

Ownerrhp@ralphslaptop

CreatedSunday, 29 May 2005

Visitor Count47

Current TemplateplainChange to: plain

Screenshots

plainjournalprojectbrown

Email Addressblog.lookmire.5749@ralphslaptop

Banner

Your NameAvon

Disable Comments☐

Inline comments☒

Email notification☐

Upload a file

Browse...Upload File

Save SettingsDelete entire blog

Appears at top of blog.

Anyone can view your blog by using this url.

The number of visits to your blog, only updated on each post.
Changing templates will remove any customization of template!

Click thumbnails to see larger view of screenshots. To change your template use "Change to" template above.

You can post to your blog by sending email to this address.
URL or filename. File must already be on server see Upload.

When you post from the web this text is shown as sender.

This stops people from adding comments.
Show comments inline on main blog page.

This will notify you via email when comments are added.

Files can then be referenced from your blog template.

Platform Independant

As with all NetWin products, SurgeMail can be run on the platform of your choice. Currently our standard builds are available for:

- Windows (NT/XP/2000/2003, 9x)
- Linux (lib6)
- Solaris Sparc (7 & 8)
- FreeBSD
- MacOSX.

This has the advantage that you can start on a platform you are familiar with, but can move your mail server to another platform with minimal effort or cost if you need to. Possible reasons could include scalability upgrades, hardware constraints or political issues.

In fact it is so easy that you can setup a live system to mirror to another host which may be another platform, then change the DNS records and you will have moved your mail server.

Which platform do we recommend?

Each platform has it's own limitations (file handles, threads etc), in addition other issues (porting problems, faulty libraries, non standard behaviour, buggy threading libraries etc all impact various systems) as a result some are better than others. Also in some cases the simple number of customers running on the same system make it a better choice. So we do recommend the following, particularly for 'large' systems: (in no particular order)

- Linux
- Windows

Systems with no known significant limitations/faults

- Windows - No known issues.
- Linux - No known issues.
- Mac OSX - No known issues.
- Solaris 8,9 sparc - No known issues.

Systems with known issues:

- FreeBSD - Threading library is faulty when used with 'fork', possibly resolved in freebsd5. This fault is not critical but is irritating, it doesn't seem to impact SurgeMail in any known way.
- Solaris 7 sparc- Limited to 250 fopen() handles, so only supports 200 mail sessions concurrently. We very strongly recommend the use of solaris 8 or solaris 9 instead.
- Solaris 8,9 x86 - Limited to 250 fopen() handles (as a result we don't support this platform)

Next: [Installs in minutes](#)

Archiving and Compliance

SurgeMail provides extensive facilities for archiving and compliance. Whether you need full SOX compliance or just want to keep track of particular accounts, SurgeMail has the controls you need. Filtering is done based on destination, source addresses, and subject. You can also select whether the archiving rule is triggered before or after any filtering that is applied, such as virus or spam filtering. This can be useful to capture the original source of viruses or spam.

SurgeMail provides facilities for:

- Network security - SSL options on all interfaces.
- Access controls - by IP, group, domain, etc.
- Authentication - via a variety of authentication modules or built in user data base
- Encryption - via SSL
- Logging and Archiving - of all transactions
- Monitoring and alerting - via separate swatch modules

Next: [Scales to any number of users](#)

Introduction

SurgePlus provides additional functionality to your SurgeMail server through the use of the SurgePlus client which your users can download and install on their machines.

Administrator Help

For SurgeMail administrator help, see <http://netwinsite.com/surgemail/help/surgeplus.htm>.

User Help

(User help and downloads are also available from your SurgeMail server. e.g. "http://your.domain.name:7080/surgeplus/").

Your email account now provides some additional facilities:

- **File Sharing** - Using your email quota you can share files by uploading them and then giving people access to the links.
Use this to
 1. Make your digital photos available on the web
 2. Create web pages without having a web site
 3. Make files available to other users
 4. Have the contents of a particular folder on your computer automatically synchronized and available on any computer you use
- **Calendar** - SurgePlus Calendar is a great planning tool. Set reminder messages for appointments and work deadlines. Since SurgePlus Calendar automatically stores information on your email server, you can access your schedule from any computer you use. Optionally share your entire calendar or just particular events with other users.

[View screen shot of Windows client interface](#)

[View screen shot of web browser interface](#)

[Download SurgeMail](#)

Upgrading from DMail

SurgeMail will detect an existing DMail configuration and create a SurgeMail configuration based on the `dmail.conf` file. In general this will successfully upgrade an existing DMail installation.

However, SurgeMail is not 100% compatible with DMail and live upgrades should be attempted with caution. (e.g. first backup your existing system so that you can back out of an upgrade)

The suggested upgrade procedure is to copy your existing `dmail.conf` file to a test system and install SurgeMail on the test system to ensure that SurgeMail will upgrade your DMail configuration. The SurgeMail installation scripts will warn you about any ini settings that it thinks it cannot convert and might be serious. If you use include files in your `dmail.conf` or have scripts that modify `dmail.conf` you should definitely hesitate as SurgeMail uses a completely different ini file format.

Also note that when DMail is upgraded external modules such as authentication modules and their databases will remain in their original location. It is important to check your `dmail.conf` and `surgemail.ini` files before deletion of any files relating to your DMail configuration.

If you find features that are missing that you really need let us know and we'll try and get them added.

The license upgrade from DMail is not free but we will give a 50% discount to any existing DMail users wishing to take advantage of the new features in SurgeMail.

Migration Details

[Migrating to SurgeMail with SurgeMail on same server](#)

[Migrating to SurgeMail with SurgeMail on a different server](#)

Migrating to SurgeMail with SurgeMail on same server.

If you want to migrate to SurgeMail and put SurgeMail on the same machine as your existing server you can still use the migration method by making the old server run on different ports. The below example is for migrating pop accounts across, but if you wish to migrate imap accounts then just need to use the imap settings instead.

Change [old pophost](#) to [old imaphost](#)

Change [old pophost nodomain](#) to [old imaphost nodomain](#)

You can put both imap and pop migration settings in surgmail.ini if you have both imap and pop accounts to be migrated.

Let's say you have oldserver running on ports 25 and 110

You now change old server to run on ports 1025 and 1110

Now you install SurgeMail on the same machine, at this point you setup the domains on SurgeMail so they match your existing domains.

Then you should stop SurgeMail, edit surgmail.ini in /etc on the windows directory.

Now you look for the domain section in surgmail.ini for the domain you want to migrate

```
vdomain address="" name="catch.netwin.co.nz"
create_user "email"
create_max "3"
create_reqd "pass_question,pass_answer"
create_repass "TRUE"
host_alias "catch.netwin.co.nz"
mailbox_path "C:\surgemail\mbox\catch.netwin.co.nz\"
quota_default "20mb"
redirect was="abuse" to="stu@catch.netwin.co.nz"
redirect was="postmaster" to="stu@catch.netwin.co.nz"
redirect was="support" to="stu@catch.netwin.co.nz"
```

So, now we add in our migration settings..

If we are migrating pop we will use the old_pophost setting

```
old\_pophost "127.0.0.1:1110"
```

So that tells SurgeMail to check on that ip and port when a user pops, SurgeMail will then log the user into that old server and collect all their mail and add them to SurgeMail's database.

If your users didn't login as user@domain on the old server and rather just logged in as user then you need to add this setting also

```
old\_pophost\_nodomain "true"
```

Now the problem is, that any mail coming into SurgeMail on port 25 for a user that hasn't been migrated will be rejected by SurgeMail as the user doesn't exist, so we need to tell SurgeMail to send that mail to the old server still. This is how we do that.

```
fallback\_relay "127.0.0.1:1025"
fallback\_check "true"
```

The first setting tells SurgeMail to send any mail for a user that doesn't exist to 127.0.0.1 port 1025

The second setting tells SurgeMail to check the rcpt exists on the old server before sending it there.

That is all!, now you leave SurgeMail in this setup for maybe a month, to give your users time to migrate after that you can remove the migration settings and remove your old server completely.

So the complete vdomain block in surgmail.ini will now look like this.

```
vdomain address="" name="catch.netwin.co.nz"

create_user "email"
create_max "3"
```

```
create_reqd "pass_question,pass_answer"
create_repass "TRUE"
host_alias "catch.netwin.co.nz"
mailbox_path "C:\surgeemail\mbox\catch.netwin.co.nz\"
quota_default "20mb"
redirect was="abuse" to="stu@catch.netwin.co.nz"
redirect was="postmaster" to="stu@catch.netwin.co.nz"
redirect was="support" to="stu@catch.netwin.co.nz"
old_pophost "127.0.0.1:1110"
old_pophost_nodomain "true"
fallback_relay "127.0.0.1:1025"
fallback_check "true"
```

You can check the status page to see how the migration is going (migrated).

USERS

Total: 70

Migrated: 35 this month

New: 10 this month

You can also check the users_yymm.rec in the surgeemail directory to see which users have been migrated, the file will contain entries like this.

1092718001 [migrated](test@ally) by (SERVER)

Migrating to SurgeMail with SurgeMail on a different server.

This is almost the same as the process outlined above, you should read that first then return here.

You simply change the settings to point to the old server again and change the MX records for the domain to point at SurgeMail, you also get your users logging into SurgeMail not the old machine. The one problem that comes into play if you put SurgeMail on a new machine is mail coming in to the old server for a user that has already been migrated due to MX records taking a while to change.

old server has ip 10.0.0.1

SurgeMail is installed on ip 10.0.0.2

User "test" logs into the SurgeMail server and migrates his account, but now a new message comes into 10.0.0.1 for "test", as SurgeMail has already migrated the account, the next time "test" logs in SurgeMail won't check the old server for mail so "test" will lose that message.. or will they?

You have two options here:

- You can use the setting [old_pophost_always](#) "true" - This tells SurgeMail to always check on the old server even if the user is already migrated, this means no mail will get lost if its delivered to the old server after the user is migrated. You can leave this setting on for several days, until the DNS's have all propogated throughout the world with the new MX entries that point to the new SurgeMail server.
- You could give the new SurgeMail server the ip of the old server and give the old server a new ip, this means that you don't have to change any MX records at all and you won't need the above setting [old_pophost_always](#) which slows the system down, this is the better option if you can do this.

Intercept migration with name translation

In the event you want to use intercept migration (for account creation - reusing old password - and mail translation) and your account old naming scheme is not directly supported by surgeemail, account name translation can be done just prior to surgeemail's login to the old server as below. This is particularly useful in the case:

- login names on old system being different from the email address (end users obviously need to change their login from old username to email address)
- Systems with really weird prefix or suffix based names which is currently not directly supported by (old_pophost_sep / old_pophost_nodomain / etc)

To use this, create a file "migration_translation.txt" of 'surgeemail login' [space] 'oldserver login' pairs for each account on your old system. This file is only loaded at surgeemail startup and is located in surgeemail root directory (as specified by g_home).

eg.

```
marijn@mydomain.com domain_marijn_suffix  
joe@mydomain.com SomeWeirdUsername
```

This page provides a starting point for the templates that can be used with Surgemail and Webmail. Some of the templates are fully maintained and supported and others are no longer being updated or have been modified by customers themselves who would rather keep the IP of their changes.

SurgeMail makes use of template to allow full customisation of the look and feel of the product. This is obviously primarily for customised WebMail webmail look and feel but can also be used with the user management and administration templates.

Panel, Surge and Smooth : Fully featured, latest template sets that have support for the latest multilanguage features of webmail, the latest antispam features and remain fully supported and updated. These are supplied and installed with the latest version of surgemail. It is highly recommended that one of these template sets is used or a custom template set is used based on one of these.



Panel



Surge



Smooth

XHtml : Very basic template set, particularly suitable for simple clients such as mobile devices and PDAs. This template set is available for download upon request.



XHtml

Marble, Iconic and Vanilla : Older templates which have not been updated to support latests surgemail and webmail features and have a somewhat out of date "look and feel". These are still available for download upon request and were previously supplied as part of the webmail standalone distribution.



Marble



Iconic



Vanilla

Customer modified templates : Many customers have given the webmail templates their own look and feel or have even written their complete own template sets. For some examples see: <http://netwinsite.com/webmail/gallery/index.htm>



Heron



Flashmail

If you are willing to share your template set or have any questions please let us know by sending us an email at surgemail-templates@netwinsite.com or webmail-support@netwinsite.com.



Customizing

Welcome to WebMail Walk Through



[CurtinWeb Mail](#)



[whooMail Center](#)



[University of Southampton](#)



[Mad-Kow.com](#)



[Tellurian Networks](#)

WebMail is a web based mail application - This will show you what other customers have done to change the look and feel of WebMail.

The html template files (*.tpl) allow you to completely customize the look and feel of WebMail to suit your particular preferences. You can easily integrate web based Email into the other services provided on your website.

As you can see from the examples in this gallery, you can just change a few details and add your own logos or you can completely redesign the entire user interface. The template files can be edited with any text editor or with a little care you can use most GUI web page design software. In the list below, a screen shot of the "list new mail" page is provided for each site. Following the screen shot is some brief detail of where the site is and who did the design. On request we can also provide a contact Email address for most of these sites so you can talk directly to other system administrators who are using this product.

WebMail Templates



[Surge Template](#)



[Smooth Template](#)

If you require more information or have any other questions please email our WebMail Support team at:

support-webmail@netwinsite.com



[WebMail Users Manual](#)



[WebMail Admin Manual](#)



[WebMail Template Guide](#)



[WebMail Walk Through](#)

Admin Manual | Advanced

Basic Installation

Overview

Windows Installation

Unix Installation

Remote FTP Installation

Advanced Installation

Setup Options

Setup Large Sites

NFS Server Setup

Upgrading WebMail

Basic Upgrade

Upgrading the CGI only

Adding New Templates

Removing Templates

Change WebMail Defaults

Default Template

Default Language

Customization

Basic Customization

Template Customization

Multi-Language Support

Other Common Links

WAPMail

Version Changes

FAQ Page

Register WebMail

Search Manual

Basic Installation

Thankyou for choosing WebMail as your Web Based Mail Client. WebMail works with a Mail Server that uses POP3 or IMAP4 and a Webserver. If you don't already have these, then consider our SurgeMail package, which includes these and WebMail.

A PDF version of the WebMail manual can be downloaded from: [webmail_pdf_manual.zip](#)

If you have any questions, please consult the [FAQ Page](#) first.

If you need to know more about any aspect of WebMail, please Email:

support-webmail@netwinsite.com

Overview

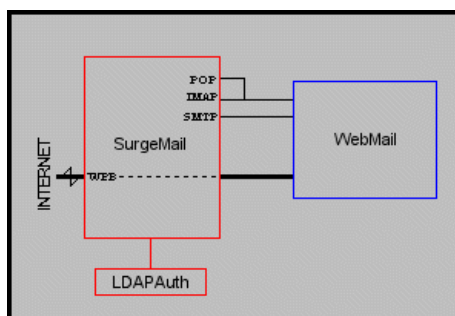
WebMail is a Web based Email Client, which means that your users only need access to a web browser to access their Email. They do not need to do any extra installation or setup of their mail account.

This allows your users to connect from anywhere in the world, using any computer, including public terminals in Internet Cafe's, not only allowing them to use WebMail as their main mail client but also read/send emails while on holiday.

The diagram below shows an example of the overall view of where WebMail fits into a mail server system. In this example the user information is stored in an LDAP database.

SurgeMail: Which is Netwin's Mail Server, which has a built in POP,IMAP,SMTP and Web Servers.

LDAPAuth: This is one authentication module you can use to interface with your database of users. There are many other authentication modules that you can use. To see the list see the URL: <http://netwinsite.com/authent/>



Windows Installation

Note: The Installation instructions are intended to help create an absolutely basic install of WebMail. For a more standard installation use the Install program included in the distribution.

Either:

Run the program install.exe

Or:

To install WebMail you must copy the webmail.exe and webmail.ini files to your CGI directory on your WEB server.

Your WEB server should have a CGI directory setup. This directory is where CGI's are installed and run. This varies from web server to web server. Below are some common locations:

"\FrontPage Webs\Content\cgi-bin"
or c:\inetpub\scripts
or \SERVER_ROOT\CGI-bin

You should read your web server documentation if you don't currently know how to setup a CGI directory.

Note: The CGI directory should have execute rights only. It should not have read rights.

The next step is to setup a templates directory anywhere on your system.

e.g. c:\webmail

First copy all the files from tpl\common to each of the other tpl\NAME directories.

Copy all the files from the 'tpl\NAME' directory from the distribution set into this directory. These files have the Extension '.tpl'. There are also some other data files that have the extension '.dat'.

Next copy all the files from img\common to each of the other img\NAME directories. Then there are the

images to be copied. Currently, the default templates presume that the images are in a relative directory '/nwimg/mail' on the web server.

ie. Physical directory:
c:\inetpub\wwwroot\nwimg\mail
\SERVER_ROOT\html\nwimg\mail

Example:

```
download the self extracting archive from netwinsite
webmail30l.exe                                     # Unzip the self extracting archive to witemp
cd \wtemp                                           # Change to temporary unpack directory

mkdir \webmail
mkdir \webmail\panel
mkdir \webmail\masterstet                         # Create a directory for templates
mkdir \webmail\masterstet\surge
mkdir \webmail\masterstet\smooth

mkdir \inetpub\wwwroot\nwimg                      # Create a directory for image files
mkdir \inetpub\wwwroot\nwimg\mail
mkdir \inetpub\wwwroot\nwimg\mail\panel           # Create a directory for image files
mkdir \inetpub\wwwroot\nwimg\mail\surge
mkdir \inetpub\wwwroot\nwimg\mail\smooth

copy webmail.exe "\inetpub\scripts"               # Copy CGI
copy webmail.ini "\inetpub\scripts"               # Copy ini
copy tpl\common\* \webmail\panel                  # Copy common template files to each
                                                    # template directory
copy tpl\common\* \webmail\masterstet\surge
copy tpl\common\* \webmail\masterstet\smooth
copy tpl\* \webmail                               # Copy docs templates etc to destination

copy img\common\* \inetpub\wwwroot\nwimg\mail\panel # Copy common image files to each image
copy img\common\* \inetpub\wwwroot\nwimg\mail\surge directory
copy img\common\* \inetpub\wwwroot\nwimg\mail\smooth
copy img\* \inetpub\wwwroot\nwimg\mail            # Copy image files

del \witemp                                         # Clean out temporary directory
```

Note: webmail.ini file must be in the CGI-bin directory with webmail.exe

Unix Installaion

Note: The Installation instructions are intended to help create an absolute basic install of WebMail. For a more standard installation use the Install program included in the distribution.

Once you have downloaded the distribution set you have to uncompress and untar it.

```
uncompress webmail10h.tar.Z  # Uncompress tar file
tar -xvf webmail10h.tar      # Extract distribution files to WebMail
                              # sub-directory
CD WebMail                   # Change to temporary WebMail sub-directory
```

Either:

run the Install program:
./install

Or:

To install WebMail you must copy the webmail.cgi and webmail.ini files to your CGI directory on your WEB server.

Your WEB server should have a CGI directory setup. This directory is where CGI's are installed and run. This varies from web server to web server. Below are some common locations:

/home/httpd/CGI-bin
or /usr/www/CGI-bin
etc.

You should read your web server documentation if you don't currently know how to setup a CGI directory.

Note: The CGI directory should have execute rights only. It should not have read rights.

The next step is to setup a templates directory anywhere on your system.

eg. /var/spool/webmail

First copy all the files from tpl/common to each of the other tpl/NAME directories. Then copy all the files from the directory 'tpl/NAME' from the distribution into this directory. These files have the extension '.tpl'. There are also some other data files that have the extension '.dat'.

Next copy all the files from img/common to each of the other img/NAME directories. Then there are the images to be copied. The default templates currently presume that the images are in a relative directory '/nwimg/mail' on the web server.

ie. Physical directory:
/home/httpd/html/nwimg/mail

/usr/www/public_html/nwimg/mail

The next step is to ensure that WebMail will run with the correct ownership. When you set up the CGI directory on your system you should have to set an ownership setting that web browser will run CGI's under.

e.g. nobody

The CGI, ini and the template directory (and template files) that you created MUST all be the same ownership, and set to something like 'nobody'

Note: On Apache, the correct owner is specified in your httpd.conf file. The default is normally 'nobody:nobody' but can be different.

Example:

After uncompressing and un-tarring the archive.

```

CD \webmail                                     # Change to temporary unpack directory

mkdir /var/spool/webmail
mkdir /var/spool/webmail/panel
mkdir /var/spool/webmail/masterset/surge
mkdir /var/spool/webmail/masterset/smooth

mkdir /home/httpd/html/nwimg
mkdir /home/httpd/html/nwimg/mail
mkdir /home/httpd/html/nwimg/mail/panel
mkdir /home/httpd/html/nwimg/mail/surge
mkdir /home/httpd/html/nwimg/mail/smooth

cp webmail.CGI /home/httpd/CGI-bin/
cp webmail.ini /home/httpd/CGI-bin/
cp tpl/common/* /var/spool/webmail/panel
cp tp/common/* /var/spool/webmail/masterset/surge
cp tpl/common/* /var/spool/webmail/masterset/surge
cp tpl/* /var/spool/webmail

cp img/common/* /home/httpd/html/nwimg/mail/panel
cp img/common/* /home/httpd/html/nwimg/mail/surge
cp img/common/* /home/httpd/html/nwimg/mail/smooth
cp img/* /home/httpd/html/nwimg/mail

chown nobody /home/httpd/CGI-bin/webmail*
chown nobody /var/spool/webmail
chown -R nobody /var/spool/webmail/*

```

Note: the webmail.ini file must be in the CGI directory with the webmail.exe file

Remote FTP Installation

You can do an installation to a hosting server even if you have only FTP access, although it is easier if you have telnet access too. Also, you must be allowed to run cgi scripts on your host. You'll have to check your service agreement or contact your host admin to find out the details of this for your host.

It is recommended that you first install WebMail on a local machine so you can get a feel for where the directories should go and get your config file setup the way you want.

If you have telnet access to your host and access to a cgi-bin directory on the host then you can just copy the WebMail download up on to your site, expand the archive and run the install script.

If you have less access than this you will have to do a manual install. It is essential to set up WebMail on a local system first to make this process manageable.

There are a number of aspects to consider:

- **Webmail.cgi and webmail.ini files.** These files should go in your cgi-bin directory. If you don't have direct access to this yourself you might have to ask your hosts administrator to put them there for you. Make sure you get all the settings in your webmail.ini right before you ask your admin to put them up. They probably won't mind the second or third time, but if you ask them to put up webmail.ini after every one of your 15 changes they might not look on you so favourably.
- **Nwimg directory.** This directory contains all the images and static HTML pages loaded by the pages WebMail displays. This directory must be in a normal webserver HTML directory. It's just like any other web pages. The kind you have probably setup before. The nwimg setting in your webmail.ini file should contain the relative url to your nwimg directory. ie, if you can view your nwimg directory in a web browser by going to <http://your.domain/webmail/nwimg> then your nwimg line in your webmail.ini should say:
nwimg /webmail/nwimg
Copy all the files from the nwimg directory on your local system to the nwimg directory on your host.
- **Workarea and templates directories.** These directories store WebMail's user files and the templates for the pages WebMail displays. These directories should be in a standard system directory that is accessible by webmail.cgi. Ideally they should not be accessible from the web, as this will mean that people can browse your users files using a web browser. If you have to put your workarea inside your webserver html directories then WebMail will still work fine but you should be aware that user files will be visible to anyone who cares to look at them. The workarea and templates settings in your webmail.ini file should simply give the system directory being used. eg.
templates /usr/imauser/webmail/templates
workarea /usr/imauser/webmail/users
Copy all the files from your templates and workarea directories on your local machine to the respective directories on your host.
- **Other webmail.ini settings.** There are a few other settings worth mentioning. Your domain setting should be the

bit that comes after the @ sign in your email addresses. eg, if your addresses are user@fish.com then your domain setting should be:

domain fish.com

You also have to specify the URL of your smtp server, and pop or imap server. You might put lines like this into your webmail.ini file:

smtp host smtp.mydomain.com

pop host pop.mydomain.com

or perhaps:

smtp host www.mydomain.com

imap host www.mydomain.com

Problem Escalation Form

Welcome to the problem escalation page, this form will let you inform NetWin product managers directly that you have been having trouble resolving an issue with our normal support staff. Hopefully this will be a very very rare event but none the less such a mechanism will occasionally be needed and will allow us to improve the service level we give. It also lets you alert our staff that you consider a problem to be very serious/urgent.

When we receive a message from this form your problem will be discussed with the relevant developers managers and support staff to ensure sensible action is being taken to resolve the problem as quickly as possible and you will receive an email to let you know what action has been taken.

Generally this form should not be used to report a new problem, your first port of call should be an email to the appropriate support email address and you should get a response within about 24 hours (excluding Sundays) support-PRODUCTNAME@netwinsite.com e.g. support-dmail@netwinsite.com, support-webmail@netwinsite.com etc..

Note: Are you in the right place? Do you personally run or manage one of our products, or are you a customer of a system that is running one of our products, if you are a customer and not running the product yourself then you probably need to find the address of the administrator of your system as we won't be able to help you directly (much as we might like to)

Your EMAIL address:

Alternate Email address: (Required, e.g. a gmail/hotmail account, some address that is not on your server)

Your FAX number: (required - if you aren't getting a response it may be because we can't email you!!)

Your Phone number

What email address did you initially report this problem to? support-dnews@netwinsite.com

What product is this regarding

How urgent is the problem Low - I need a fix within the next month or so

What platform are you on Windows NT

Please describe the problem and give any additional information we might need.

Include the relevant config file surgemail.ini etc, for cgi products include the relevant .ini file and the text of the error if possible in the following text box..

If we can get telnet or ssh or pc anywhere access to the system that may help in resolving the problem quickly. ([See this page for submitting passwords securely](#))

You can expect a response within 24 hours, if you don't get one fax us on +64-6-353 7359

Is it possible to stop all spam without losing emails?

First, systems that don't work, and why:

- **Accepting all email** and let the customers figure out which ones are real: No good as humans miss identify spam/not spam at a high rate 1-2%
- **Filters**: Content filtering also starts to fail at a rate of about 1% without continuous tuning. Partly because an email 'about spam' looks very similar to a spam message.
- **Sorting/Tagging**: Here the computer/human are both doing the sorting, the problem is it doesn't improve the accuracy, only the speed, so again failures occur and real email is lost.
- **Simple SPF**: The spammers can always send while forging domains that don't have SPF records.

We believe the best approach is two fold, first using SPF. With the default SPF setting, 90-95% of spam is blocked before it reaches your users. This is important as it saves them time, and means that they can spend sufficient time examining any messages that do get through to make a good decision and not accidentally delete real email.

Second, turn on friends mode for anything rating above '4'. This will send anything suspect back to the sender with instructions for the sender to bypass the filter.

Now lets examine how SurgeMail can stop spam if your settings are adjusted correctly.

- **Default SPF checking** - by checking domains that don't have an SPF record with a default SPF record 95-98% of spam is stopped before it enters your system.
- **Friends system** - this stops any email getting to you that is not 'from' a person you know.
- **Rule file** - a rule file marks messages which 'might' be spam, you can set friends to only intercept messages that look like spam so emails from most people will never get even 'one' friends bounce.
- **Private addresses** - By setting a private address, you can bypass all the spam/spf filtering when you need to, for automated robots, banks, etc. anything where you need to be sure you get the email and you trust the sender with your 'private' email address.

What you need to do

1. Install SurgeMail or SurgeWall in front of an existing mail server. :-)
2. The surgemail administrator should upgrade and press the button to check config settings (on the global settings page)
3. Go to user self management page, and define a 'private' address.
4. Turn on friends mode, for anything rating above '4'

Lets now examine the problems you might still face.

Putting your address on a web page.

You can still do this, but consider using a web form instead as it will reduce spam. We provide a simple utility to do this if you don't already have one: <http://netwinsite.com/easyforml>

Filling out a web form at a trusted web site.

Any trusted reputable site (e.g. bank, airline, online retailer...) will have a privacy statement on any form where they request your email address, if you are sure they are legit, then give them your private email address, user--PRIVATEWORD@your.domain, this will bypass your friends checking etc. and ensure you get anything from them. If they are not trusted, then use your normal email address, and if necessary add an exception rule in user settings to ensure you get the email from them.

Joining a mailing list or posting to a news group

Definitely use your normal address, don't use the private variant or you will get spam!!!

SPF

SPF (Sender Permitted From) is a new mechanism which allows you to define what IP addresses are permitted to send mail 'from' your domain, this will stop spammers from pretending to send message from your domain.

What you should do?

- [Define an SPF text record](#) in your DNS entry for your domain(s) (this is not required but is a good idea)
- If using surgmail, update to SurgeMail 2.0e or later and add these settings:

```
g_spf_mode "strict"
g_spam_block "true"
g_spf_rewrite "true"          (optional, or you can turn it on per domain or per user)
```

What setting makes it block messages ?

Nothing is blocked when you set `g_spf_mode "strict"` so that is always safe. When messages are blocked they are blocked using the SurgeMail 'allow' system so that the sending user (if they are a real human) can fix the problem without intervention from an administrator.

To actually block spam you must do one of these:

- 1) Set `g_spam_block "true"` (that sets it on for everyone by default)
- 2) Set the domain setting `spam_block "true"` (which sets it on for everyone in a given domain)
- 3) In the web user settings, login and click on SPF on the left hand side, then set the option box at the top to "True"

Doesn't SPF rely on the senders creating spf records ?

No, in strict mode surgmail makes up an spf record for all incoming domains so it works for everyone. When the made up spf record fails (which is rare) surgmail then provides other checks and mechanisms so real email can still get through.

What about 'DomainKeys' ?

DomainKeys is a cryptographic solution which is similar to SPF, in general, SPF does everything that domainkeys does, but without the extra load and complexity of cryptography. We recommend you also use domainkeys tests (available in the latest versions of surgmail) for incoming email to cope with the two large providers who refuse to define SPF records (apparently for political reasons).

Also see [SurgeMail Spam Prevention Guide](#)

Why/How will SPF stop spam?

There are two types of spam, legitimate businesses sending email from real domains to people who haven't asked for it, this type of spam is annoying, but trivial to filter with simple rules and RBL databases. And most businesses are learning not to do this as they rapidly find themselves cut off from the customers they do want to talk to. This type of spam will continue but at a relatively lower level, it isn't really a problem.

The second type of spam is the problem, it's sent by people who use fake 'from' addresses and domains, via multiple IP addresses and virus mail slaves, meaning each email comes from a new IP address, each email is written specifically to evade the filters, and new variants are written each day. These mails are more or less impossible to filter. However, this second set is trivial to block with SPF!!!

SurgeMail and SPF (Cool features)

- Full implementation of SPF, SRS and Macro's, all controlled with simple config settings (**Install version 2.0 or later**)
- Built in test page <https://localhost:7025/?cmd=spf> for testing SPF processing on sample addresses.
- Additional 'strict' settings to block spammers while letting through most legit email even if SPF records don't exist.
- Additional 'allow' mechanism so false positives can be corrected without administrator intervention
- Internal long term IP database of known addresses, known spammers, and known non spammers, linked to the 'aspm' system.

Turning on SPF with SurgeMail

Add the setting 'g_spf_mode', set it to "block", or "stamp", or "strict"

g_spf_mode	Meaning
block	Bounce emails that come from a domain with a valid SPF record if the SPF return code is 'FAIL' By default the 'allow' mechanism will be used so nothing is actually blocked unless you set the domain or user default to 'block'. The domain setting to make it actually block by default is spam_block "true" . Note: "block" mode will not block email from domains with no SPF record, use "strict" mode instead.
stamp	Do the SPF processing and add a header to the message, which is then used by the filter to score the message more accurately as spam
strict	<p>This setting is unique to SurgeMail, it means if there is not a good 'SPF' record then SurgeMail applies a default rule "mx/16 a ptr:%{d2} -all" settable with g_spf_default which basically means is the person sending this message from a mail server that accepts messages for this domain (or close to it) or if I lookup the name of this ip number, does it match the domain name. Then SurgeMail checks if it's an ip address it knows about historically, and if still can't find it, then it blocks the message. This will block some real email, but it will block a 'lot' of spam. By default the 'allow' mechanism will be used so nothing is actually blocked unless you set the domain or user default to 'block'. The domain setting to make it actually block by default is spam_block "true"</p> <p>Set strict for ASPAM scoring only</p> <p>If you just set strict mode, but don't set spam_block "true" or g_spam_block "true" then the messages will be stamped only, the aspm rules will notice this and score them accordingly, so it is useful to set strict mode without setting anything to actually block messages.</p>

When SurgeMail does bounce a message (which will be rare anyway due to the other checks) it will give an email address that the sending person can send to to 'allow' their ip address. Then that user will be able to send without problems. The entire net block is enabled x.y.z.* so related mail servers won't get problems.

The following additional settings may be useful too:

g_spf_domain "main.domain.name"

This is the domain spf will use for the unblock message set this if the default domain it chooses is not appropriate. Make sure email addressed to this domain will arrive on this server (e.g. add an mx or a record if needed)

g_spf_rewrite "true"

This enables SPF/SRS rewriting of from addresses on forwarded email, which is crucial if other servers are obeying SPF records, you only need this setting if you allow users to forward email to other servers. If you have **multiple incoming mail** servers copy the file **srs.secret** to all of the servers in question (it should contain a random number.

g_spam_allow_rbl "true"

This applies the same 'allow' mechanism to RBL/ORBS lookups, which means it's much safer to use a 'deny' action instead of the softer 'stamp' action.

g_spam_allow_msg "||reason||, send to ||allow|| then resend your message."

This lets you change the format of the 'allow' error.

g_spam_allow_known "true"

This is good to set as it means only new 'unknown' ip addresses are blocked for SPF reasons. so it's not a transient spammer"

g_spf_skip_from "noreply@*.paypal.com"

Use this setting to selectively allow from addresses from badly behaved systems which routinely come from random ip addresses,

like paypal. As long as you keep this list short and fairly specific it will be very difficult for a spammer to make any useful use of it.

```
g_spf_rev_skip "*.ebay.com"
```

Setting to skip spf checks if reverse ip matches, e.g.

```
g_spf_rev_skip "*.yahoo.com"
g_spf_rev_skip "*.ebay.com"
```

Use to define any badly behaved forwarding sites (sites that forward email for many domains but don't correctly use SRS rewriting rules.

```
g_spf_share "other.mx.host"
```

Use this setting so when an 'allow' message is received and is valid it is then forwarded to other mx hosts so the information is shared. You must also copy the srs.secret file to all mx servers.

g_spf_default

This lets you over-ride the default rule which is applied when a domain has no spf rule. The default is:

```
mx/16 a ptr:%{d2} -all
```

However the 'd2' is replaced with 'd3' for country level domains. So generally this default is best left alone.

Settings which control what SurgeMail does with an spf failure

Setting	Default	Description
g_spam_block	false	If set, then if spf checks fail message is blocked
g_spam_block_gateway	false	If set, and if message is being gatewayed, then if spf checks fail messages is blocked
spam_block	false	If set, then if message is for this domain, then if spf fails then message is blocked.
spam_block (authent response)	None	This is the authent response, which can be set to one of three values 'not set', 'true', 'false'. 'if not defined' = Message is blocked or not based on global and domain rules above. 'true' = If spf fails, message is blocked ignoring above global and domain rules. 'false' = Messages for this user are never blocked.

Gateways & Filters and secondary MX hosts

SPF must be implemented on the servers that **receive email from the outside world**, so if you have filters in between the internet and your real mail server, then you need to **enable spf on the filter system**.

The same applies to secondary MX hosts, but in this case you must do one of two things so that the messages from the secondary mx host are accepted by the primary host, e.g.

- With SurgeMail version 2.0h or later use g_spf_skip "x.x.x.x" where 'x.x.x.x' is the ip address of your other mail server(s) (a comma seperated or wild card list).
- Turn on g_spf_rewrite "True", which means anything that is accepted for delivery by the secondary mx is from then on immune to spf checks from 'any' server.
- On the primary server list the secondary server as a trusted host e.g. 'g_verify_mx_skip x.x.x.x' where 'x.x.x.x' is the ip address of the secondary server.
- Use g_spf_share "other.host1,other.host2" on each server so they can share allow information between themselves.

NOTE: Copy the file **srs.secret** between all cluster/mx servers so it matches on all systems, this file contains a random number which is used to verify srs and allow emails. This file will only exist once you start using SPF, it resides in the `g_home` directory.

tellmail spf_export file.name
tellmail spf_import file.name

Exports the list of 'known' ip addresses to a file suitable to copy to another system, use this when adding an incoming mx gateway if you want it to already know about all existing 'trusted' ip addresses. The file name is relative to the surgemail home directory, not your current working directory!

Authent Module support

Your authent module must support an extra string field 'spf_block'

spf_block, valid values are 'true,false,blank'
true = block
false = don't block
blank = system default.

+OK user@here.com 0 spf_block="FALSE"

Debugging

See the file **allow.log** to see why messages are let through or blocked when using these settings.

How to define an SPF record for my domain

Start with something like this, replace 1.2.3.4 with the ip address of your mail server.

v=spf1 ip4:1.2.3.4/24 -all

That's probably all you need. You can send from any address 1.2.3.0-1.2.3.255 which will probably be fine for most purposes. It's simple, and efficient. If you want though you could add a little more. But remember the more you add, the slower it is to process, and the more likely you will make a mistake. Here is a good basic rule.

v=spf1 ip4:1.2.3.4/24 a mx -all

Token	Explanation
v=spf1	Version of SPF syntax
ip4:1.2.3.4/24	Allow any ip address 1.2.3.0-1.2.3.255 (change to match your own mail server ip address)
a	Allow any ip which matches the IP address of this domain (doing a simple 'a' lookup)
mx	Allow any ip which matches the IP address of a mail server that accepts incoming mail for this domain.
-all	Block any mail from an ip other than those listed above

How do I check my spf record for my domain

The best way is to use the SPF test page in surgemail (Spam Control - SPF Settings - Test SPF Records). This will allow you to test any sending ip address + from address + spf record combination. This allows you to:

- 1. Check your new spf record before configuring your DNS server
- 2. Test any domain to see whether an spf record has been setup, and setup correctly

eg sending mail from valid (216.65.3.228) and invalid (1.2.3.4) ip address for surgemail-support account on domain netwinsite.com :

spf: Start ip=216.65.3.228 from=surgemail-support@netwinsite.com
spf: lookup (txt) (netwinsite.com) --> (v=spf1 ip4:216.65.3.0/16 ip4:210.54.44.0/16 a mx -all)

```
spf: SPF txt record found (v=spf1 ip4:216.65.3.0/16 ip4:210.54.44.0/16 a mx -all)
spf: +++ Proccessing Token=(v=spf1) Parameter=(spf1) cidr=32 +++
spf: Processed token (v=spf1) - continuing .....
spf: +++ Proccessing Token=(ip4:216.65.3.0/16) Parameter=(216.65.3.0) cidr=16 +++
spf: spf: cidr(216.65.3.228, 216.65.3.0, /16) Matched Mask
spf: Last token {ip4:216.65.3.0/16} (res=PASS)
```

```
spf: Start ip=1.2.3.4 from=surgemail-support@netwinsite.com
spf: lookup (txt) (netwinsite.com) --> (v=spf1 ip4:216.65.3.0/16 ip4:210.54.44.0/16 a mx -all)
spf: SPF txt record found (v=spf1 ip4:216.65.3.0/16 ip4:210.54.44.0/16 a mx -all)
spf: +++ Proccessing Token=(v=spf1) Parameter=(spf1) cidr=32 +++
spf: Processed token (v=spf1) - continuing .....
...
spf: spf: cidr(1.2.3.4, 216.65.3.228, /32) No match
spf: Processed token (mx) - continuing .....
spf: +++ Proccessing Token=(-all) Parameter=() cidr=32 +++
spf: Last token {-all} (res=FAIL)
```

An alternative way to check your SPF record is using nslookup or dig as per:

```
[root@linux]# nslookup
> set type=txt
> netwinsite.com
Server:      10.0.0.25
Address:     10.0.0.25#53
Non-authoritative answer:
netwinsite.com text = "v=spf1 ip4:216.65.3.0/16 ip4:210.54.44.0/16 a mx -all"
```

Other SPF resources:

- The [SPF Home Page](#)
- The [SPF wizard](#) for generating rules

The rest of this page summarizes some features of SurgeMail.

- [Integrated mail server and web based email](#)
- [Spam and virus free email](#)
- [Web based administration](#) (user, domain and server)
- [Unique Mirror system](#)
- [Full SSL secure support for all protocols](#)
- [Interface to existing user databases](#) (OBDC, MySQLI, LDAP, etc or use built in database)
- [Scales to any number of users and domains](#)
- [Runs on platform of choice](#) (Windows, most Unix versions)
- [Installs in minutes](#) and unparalleled after sales service
- [Cellphone SMS/Text messages](#) via SMSGate (FREE)

ORBS

Open Relay database check

SurgeMails integrated and flexible open relay database checking ([g_orbs_list](#)) can be used enforce a servers blacklisting or whitelisting in one or more ORBS databases. In addition this can be used to mark messages with a header which can then be taken into account in the SmiteCRC"SpamDetect rating" calculation. An ORBS database is simply a DNS server that returns a positive response if a server is listed in the database. A variety of services are available online that can maintain blacklist databases. Normally you would maintain your own whitelist database that overrides the blacklist listings.

eg 1 - A simple deny mail from blacklisted servers could be achieved with:

```
g_orbs_list name="relays.ordb.org" action="deny"
```

eg 2 - A smarter setup with exceptions for certain IP ranges and a whilelist exception database, a blacklisted deny database and with useful header based tagging could be achieved as follows:

```
g_orbs_exception "127.0.0.*,12.34.56.*"
g_orbs_list name="mywhitedatabase.none" action="accept"
g_orbs_list name="relays.ordb.org" action="deny"
g_orbs_list name="bl.spamcop.net" action="stamp" stamp="spamcop, http://spamcop.net/w3m?
action=checkblock&ip=||remoteip||"
```

eg 3 - To use the output of header based ORBS stamping in the Aspm SpamDetect calculation the following could be used:

```
g_orbs_list name="relays.ordb.org" action="stamp" stamp="open relay"
g_orbs_list name="my.dialup.databse.none" action="stamp" stamp="dialup"
```

These entries have the following rules in filter.rul. If you used your own stamp text you would place appropriate entries in the local.rul file.

```
if(rexp_case("X-ORBS-Stamp", "open relay")) then
call spamdetect(4.0, "Sender's IP was on an open relay RBL")
end if

if(rexp_case("X-ORBS-Stamp", "dialup")) then
call spamdetect(4.0, "Sender's IP was on a dialup RBL")
end if
```

SPF

SPF is a recent mechanism that provides for verification that the sender is actually allowed to send from the domain that they say they are sending from. This has the potential to reduce a very large amount of worldwide spam. For more information see <http://netwinsite.com/spf.htm> .

Virus Protection

SurgeMail has a variety of mechanisms for integrating with [commercial and free products](#) .

- [Integrated and efficient Avast scanner](#) (windows / linux only)
- [Efficient external scanners](#)
- [External SMTP scanner](#) ("virus wall")
- [Arbitrary command line scanner](#) (deleting message or return code)

Any of these mechanisms can be used but it is recommended that Avast is used as this is closely integrated with surgemail, is efficient and is less prone to errors under load.

Integrated and efficient Avast scanner (windows / linux only)

Avast is an integrated, efficient and cost effective virus scanner produced as third party product developed and maintained by [ALWIL Software](#).

[Click here](#) to install Avast

Avast is automatically licensed as part of your surgemail license.

Under windows avast is integrated using a DLL that is loaded if required and on linux avast is integrated using a "g_virus_filter" external vpipe scanner. A variety of performance and usage statistics are available on the advanced status page / tellmail status output.

Efficient external scanners

SurgeMail has a interfaces that allows external scanners to be efficiently run (as a daemon / running executable) so that surgemail does not have to run an executable for each message processed. These require the virus scanner to support the required specialised syntax, most commercial scanners do not support this syntax.

- **vpipes virus filter** ([g_virus_filter](#))

Originally setup as surgemail's primary antivirus interface, but is also very useful to integrate custom spam prevention tools. This allows you to run one or more virus checkers or filters that takes commands on stdin and response on stdout using simple command line syntax:

- Surgemail send filter : "nn scan <message file> <envelope file> "
- Filter responds: "nn CLEAN comment " or "nn REJECT comment"

where: nn would be a numeric command identification, message file a file containing the email message including all headers and envelope file a three line file containing RCPT, MAIL FROM address and sender's IP address as follows:

-- start of file --

To: <marijn@destdomain.com>

From: <marijn@sourcedomain.com>

IP: 127.0.0.1

--end of file --

eg.

d:\surgemail\rav8\bin\ravdmail.exe

1 scan c:\temp\test.txt c:\temp\test.env

1 CLEAN c:\temp\test.txt that was a good message

2 scan c:\temp\test.txt c:\temp\test.env

3 REJECT c:\temp\test.txt contained a virus

- **FProt daemon interface**

Surgemail will interface directly to the fprot daemon as [described](#).

External SMTP scanner

External smtp scanners (sometimes called "virus wall" scanners) can be used by surgemail. If this kind of scanner is used it is strongly recommended to let surgemail receive the mail from the internet, and then have surgemail pass all mail through the external virus scanner before delivering this locally. This will allow you to make full use of surgemail's spam prevention measures - primarily measures such as SPF, RBL etc that use sender's IP address. If mail is passed through the external scanner BEFORE it gets to surgemail, surgemail will see the external scanners ip address for all inbound mail.

This is configured using gateway rules as follows:

```
g_gateway domain="*" to="scanner.ip"
user="" pass="" relay="false" check=""      => pass all mail to scanner*
sms="" local="TRUE"                        => deliver local deliveries if from scanner
g_gateway_ifnot "scanner.ip"               => deliver remote deliveries if from scanner
g_gateway_ignorewild_ip "scanner.ip"       => allow scanner to send outbound scanned messages
g_relay_allow_ip "scanner.ip"
```

* **notes:** local="TRUE" means "only accepts mail for local domain if the account exists" and relay="FALSE" means "only accept outbound mail if using smtp authentication or other relay enabling setting" - this is called "restricted relay" in the web admin interface)

Mail getting gatewayed to specific domains through the use of the standard gateway rules can also be scanned. To configure this make sure you have the external smtp scanner setup as described above and the wildcard scanner g_gateway rule is before any gatewayed domains to make sure gatewayed mail is sent to the scanner before the normal gateway rule is matched.

```
g_gateway domain="*" to="scanner.ip" user="" pass="" relay="false" check="" sms="" local="TRUE"
g_gateway domain="gateway.domain" to="dest.server" user="" pass="" relay="" check="TRUE"
sms="" local=""
```

Arbitrary command line scanner (deleting message or return code)

SurgeMail has the ability to integrate with a variety of other external scanners. These are all less efficient as a separate external process gets executed for each message that passes through the system

- **g_virus_cmd**

If defined the mail server will extract MIME parts in a multi part message and run the virus scanner over the extracted file. The command line can include \$FILE\$ which will be replaced with the actual file name of the extracted part. An intelligent cache is used so mailing lists, etc, will not require running the virus scanner on every message sent.

If you set this to "do_not_run" then SurgeMail will extract the MIME parts but not actually run any program, some virus scanners scan all files on the system so the file is deleted magically and SurgeMail will notice and bounce the message.

If your scanner supports the returning of return codes if a virus is found then you should use g_virus_cmd_codes with this setting as this is more reliable than having to detect if a file is deleted and also means also will work on viruses in archives which a lot of scanners won't delete.

- **g_virus_cmd_codes**

This lets SurgeMail listen to the return codes from g_virus_cmd and if the return code matches one specified in this command then it will assume its a virus and reject the message. This is often more reliable than detecting only by using g_virus_cmd as some virus scanners do things a bit differently. Also a lot of scanners won't delete archives containing viruses so this is the best way if your scanner supports it.

How do I configure surgemail using "Scanner X"

Many scanners they can be integrated in one of several ways dependent on how the scanner is configured. You should double check the virus scanning documentation to check the error codes given in the examples is still up to date.

Virus Scanner	Platform	Scanner (surgemail.ini entry)
---------------	----------	-------------------------------

Avast Recommended!	Windows	Use web admin tool to enable
	Unix	Available on Linux only
	<p>Comments: Highly recommended, fully integrated for ease of installation and performance. Note: "Avast! for SurgeMail" based on ALWIL Software antivirus technology.</p> <p>To enable go to the Avast section of the web admin tool (see details), but first purchase the 'Avast' license option from NetWin and re-activate your registration.</p>	
ClamAV	Unix	<p>Note: we recommend avast for any serious load, it is more efficient and more reliable</p> <p>The best way to run ClamAV with SurgeMail is to use the scripts provided by one of our users. Instructions and downloads can be found here for the scripts (SCAVS) http://www.inoc.net/~dev/surgemail/scavs/</p> <p>Or you can use the following lines but on busy systems you could run into problems with too many channels being tied up waiting for ClamAV to finish.</p> <p>Check the path below using 'whereis clamdscan' it may be /usr/bin/clamdscan</p> <p>Next in /etc/clamd.conf set clamd to run as user 'mail' and then restart clamd, you will need to set the ownership of the clamd directories to 'mail' too, e.g.</p> <pre>chown -R mail /var/clamav (do this for all the paths in clamd.conf) cd /etc/init.d ./clamd stop ./calmd start</pre> <pre>g_virus_cmd "/usr/local/bin/clamdscan --stdout --no-summary --remove \$FILE\$" or g_virus_cmd "/usr/bin/clamdscan --stdout --no-summary --remove \$FILE\$" g_virus_cmd_codes "1"</pre>
	<p>Comments: Free!, opensource,supports many UNIX platforms Set up the daemon (read clamav instructions)</p> <p>If you have a ramdisk then clamd will go faster if you make it use the ramdisk for it's temporary directory.</p> <p>YOU MUST SETUP THE DAEMON FIRST, or it will still run but be very slow and cause problems for SurgeMail.</p>	
Command Antivirus	Windows	*
	Unix	g_virus_cmd "/usr/bin/csav -delete \$FILE\$"
	<p>Comments: We recommend avast for any serious load, it is more efficient and more reliable</p>	
F-Prot (daemon scanning)	Windows	N/A
	Unix (Daemon Mode, not available for Windows)	g_virus_fprot "11200"

	Comments: Full daemon support, very fast, the setting is the port the daemon runs on. You must also install the daemon for this to work, see g_virus_fprot	
F-Prot (command line scanning)	Windows	g_virus_cmd "c:\progra~1\fsi\f-prot\fpcommand.exe \$FILE\$ /auto /delete /archive /silent" g_virus_cmd_codes "3,6,8"
	Comments: Command line scanning, We recommend avast for any serious load, it is more efficient and more reliable	
McAfee	Unix	g_virus_cmd "/usr/local/uvscan/uvscan --analyze --mailbox --mime --program --secure --unzip --noboot --delete \$FILE\$" g_virus_cmd_codes "12,13"
	Comments: instructions from a user, We recommend avast for any serious load, it is more efficient and more reliable	
Norton Antivirus	(demand mode)	g_virus_cmd "c:\program files\navnt\navwnt.exe /noresults \$FILE\$" g_virus_cmd_sleep "3000"
	(always scanning) real time scanning	g_virus_cmd "do_not_run" g_virus_cmd_sleep "3000"
	If you have Norton's setup already and it scans files as you access them then use the real time scanning option otherwise use demand mode. Norton is best avoided for mail servers! We recommend avast for any serious load, it is more efficient and more reliable	
Sophos	Windows	g_virus_cmd "c:\programs\Sophos SWEEP for NT\sav32cli -s -nc -remove -archive \$FILE\$" g_virus_cmd_codes "3"
	Unix	g_virus_cmd "/usr/local/bin/sweep -s -nc -remove -archive \$FILE\$" g_virus_cmd_codes "3"
	Comments: A very well known and respected virus checker that is available for a wide variety of platforms, is well suited for SurgeMail with its on demand scanning utility - Sweep. We recommend avast for any serious load, it is more efficient and more reliable	
TrendMicro	Windows	g_virus_cmd "c:\sysclean\vscantm.bin /nbpm /D /Q /NM /NB \$FILE\$" g_virus_cmd_codes "1"
	Instructions: Use this link to find out how to setup Trend for command line scanning. http://kb.trendmicro.com/solutions/search/main/search/solutionDetail.asp?solutionID=17058	
NOD32	Windows	g_virus_cmd "c:\Program Files\ESET\ESET NOD32 Antivirus\ecds.exe /no-log-console /clean-mode=delete \$FILE\$"
	Comments: We recommend avast for any serious load, it is more efficient and more reliable	
	Windows	

RAV antivirus for DMAIL		g_virus_filter cmd="c:\surgemail\rav8\bin\ravdmail.exe"
	Unix	g_virus_filter cmd="/opt/rav/bin/ravdmail" type=""
	Comments: RAV is no longer being sold as a product (this is here for the benefit of existing RAV users)	

* This configuration has not been explicitly tested. If you have experience with a virus scanner that is not listed here please let us know so we can help others in their server configuration efforts.

Aspam Overview

Aspam is a more efficient and more accurate spam recognition system to replace the SmiteCRC spam recognition system. For end users it is functionally equivalent to SmiteCRC spam recognition (each message gets a score and administrators and end users can filter email based on this score).

Rather than reading this page you should probably be reading [Spam Prevention Guide - spam.htm](#)

Aspam uses the following mechanisms to recognise spam.

- MFilter rule set - rules which recognise most common spam, about 60% effective on untrained data. Updated automatically from our web site.
- Aspam auto generated rules, about 99.5% effective on trained data, about 40% effective on untrained data
- Poly, multi symbol statistical matching, about 95% effective on untrained data.
- Aspam and Poly are both 'auto trained' by users submitting to two special local addresses (one for spam that was not recognized - isspam@mydomain.com, and one for non spam that was incorrectly identified as spam - notspam@mydomain.com) This training feature should not be over emphasized, it allows easy fine tuning for localized spam, but don't go over board trying to train it with thousands of messages, that isn't necessary..
- In addition network checks are made to confirm the identity of sending users, these are also very effective.
- Aspam and Poly 'base' knowledge bases are automatically downloaded from NetWin so you don't need to get thousands of training messages before the rules start working. (But matching improves of course as local users send training messages)
- Catcher addresses - Which are local addresses which are 'hidden' on your web pages and in other public forums any emails to these addresses (that you list) are recorded as spam and used to train the system. This defeats people who send emails using robots to every address on your web pages.

Files used by system

```
aspam_bad (directory of training messages, collected from isspam@local.domain address)<>
aspam_bad.rul (aspam rule file automatically generated from messages in aspm_bad directory)
aspam_good (directory of training messages of 'good' non spam messages)
aspam_good.rul (aspam rule file automatically generated from messages in aspm_good directory)
aspam_rules.txt (Base aspam rule set from Netwin - auto downloaded)
aspam_mfilter.txt (Base mfilter rule set from netwin - auto downloaded and auto updated)
local.rul (Your place to add mfilter rules to adjust scoring)
aspam_pgood.dat, pbad.dat (Base poly word matching rules, auto downloaded from Netwin)
aspam_words.txt (English dictionary used by aspam when choosing which words are rare)
poly_good.dat (Poly word knowledge base (binary file) created from aspm_pgood.dat and local sample messages)
poly_bad.dat (as above)
```

Tellmail commands:

tellmail aspam_retrain = Rebuilds the rule and poly information based on the aspm_good, aspm_bad directories

Aspam details

- Aspam-URL database (collected via the training addresses) of known 'bad' URLs, if these are found in messages the message is very likely spam. Any URL found in the 'notspam' database is 'whitelisted' automatically.
- Aspam-known External IP addresses - these are also recorded from the training messages and remembered and used for scoring
- Known words - a list of rare words are found in each spam message, if the same list of words is found in a new message it gets a high match (for spam or not spam depending on the original)

How to turn on aspam

In surgmail.ini add **g_spam_internal "true"** and **remove** the line **g_virus_filter cmd="smitecrc.exe" type=""**

You can also do it with the web admin tool, eg:

Open the web admin,
upgrade to 1.8
click on 'SmiteSpam' down the left hand side,
set "Enable SmiteSpam" to 'false',
press 'Save'
click on Aspam on the left hand side,
set 'enable aspam' true
set 'Verify sender is in MX record' true

set 'Check from user exists' true
set 'If score is above this, add spam rating to subject (Spam: ****) e.g. 5' to 5
set 'Allow users to specify specific spam features' TRUE
Fill in 'Addresses on web pages that shouldn't get any email (robot bait)' if you have any.
press 'SAVE"
(Restart SurgeMail)

Recommended Settings for ASpam

Check senders are valid (optional, does help but also slows down incoming SMTP)

g_badfrom_check "TRUE"
g_badfrom_stamp "TRUE"
g_badfrom_from "postmaster@YOUR.DOMAIN.NAME"

Check sender is sending from the right mx host if possible. (this is the best rule!)

g_verify_mx "true"

Mark spam messages in the subject (optional, some people like this some don't)

g_spam_subject "6"

Auto train addresses that spammers send to (replace with your addresses)

g_spam_catcher "user1@domain.com,user2@comain.com"

Known faults - gotcha's etc

- In 1.8b2 the auto train feature will get confused and auto train messages sent to the iss spam address, we recommend not using the auto train setting.

Mfilter Rule Syntax

Design Goals:

- Fast processing of incoming messages.
- A simple, clear, syntax so that rules can easily be understood and modified
- Enough power/flexibility
- Incorporate regular expression matching to give real power.
- Not to create a new elaborate language if possible.

How to configure rules:

- Simply create a file called **mfilter.rul** in the **SurgeMail work area** (as defined in SurgeMails config)
- Use the test command via the SurgeMail admin interface to check that your filter works as expected
- Please note that "local.rul" should be used for adding scoring for ASPAM not mfilter.rul.

Tracing problems

If you have problems getting mfilter to run you can use these two settings in surgemail.ini, they will provide logging to show exactly what is going on.

```
g_mfilter_trace "true"
g_mfilter_noisey "true"
```

then examine 'mail.log' after sending in a test message.

Syntax Of mfilter.rul File

There are 6 valid statements in a rule file:

```
Assignment
Action
if (Conditional_Expression) [and (Conditional_Expression)...] Action
else
end if
call built_in_function()
```

Assignment

```
$variable_name = "quoted string" [+ "quoted string" [+ $variable ...]]
$variable_name = function()
```

Action

```
accept "reason" | bounce "reason" | drop "reason" | forward "user@domain" | then | setflag("flagname") |
clearflag("flagname")
```

Conditional Expression (if, else, end if)

```
Any pre-defined function, e.g. isbinary()
isin("subject","free pictures")
Numeric comparisons, e.g. lines()>100
Simple NOT operator, e.g. if (!isbinary()) reject "Only binaries allowed here mate!"
Calculations are NOT permitted, e.g. lines()+10 would fail
```

Recipients block for processing individual recipients

A single mail message may have many recipients, and in many cases the actions of your spam filter should vary depending on the recipients (you might, for example, want all messages to your account to get through even if the same message would be blocked if sent to any other user).

The recipient block (recipients...end recipients) is processed once for each recipient of the message.

Inside the 'recipients' block there is a dummy variable defined 'recipient' which is the specific recipient in question.

All the action's (except, bounce, drop) refer to the recipient only, not to the entire message, so when one of those actions that normally terminates message processing is encountered (accept, bounce, drop, etc) instead the action is applied only to that recipient and the recipient block is restarted with the next recipient defined.

(Example of mfilter rule to do processing 'per recipient')

```
recipients
  if (isin("recipient","manager@this.domain")) accept "Always accept for me      so spammers can talk to me"
  if (isin("recipient","sales@your.domain")) then
    if (isin("subject","order")) then
      # Make a Duplicate of sale order
      call forward_cc("sales_copy@your.domain")
    end if
  end if
end recipients
```

Miscellaneous

Line Continuation

Lines can be continued by ending the line in a '\' character

Quoting Strings

All strings and header names should be within double quotes, sometimes you may get away without doing this, but we don't guarantee this will work in future. e.g. use: exists("Supersedes") not exists(Supersedes); quotes can be escaped in the usual way, e.g. "This \"Word\" has quotes around it"

Assignments

Assignments are processed at compile time, variables DO NOT exist at run time. Do not think of this as a programming language, but rather as a list of rules that are processed with each incoming message. Real run-time variables only exist in the form of the ifflag("xxx") function and the setflag("xxx") action.

For example, the following is NOT VALID, as the assignment is processed before the rules are run. The rejection would always read "big message"

```
$fred = "small message"
if (lines(>100) then
  $fred = "big message" (this will not work as expected)
end if
reject $fred
```

Odd stuff

The statement '**do_bounce_fast**' should appear at the end of your mfilter.rul file, and it is used by the rexp_fast() rules. **rexp_fast** acts just like rexp() but it is much faster because it searches the message once for all of the rules in question, each rule must start with two simple non 'regular expression' characters. This enables mfilter to generate a hash table of all the regular expressions it's going to search for and then it can efficiently apply only the ones that appear to match as it runs through the message. Also rexp_fast includes the score to apply if the message matches the rule.

Actions & Commands

Actions

[accept](#) "reason" (Terminates processing)
[bounce](#) "reason" (Terminates processing)
[reject](#) "reason" (same as bounce)
[forward](#) "reason" (Terminates processing) (redirect is a synonym for this action)
[print](#) "reason" (Prints debugging line to log file mail.log)
[setflag\("flagname"\)](#) "reason"
[clearflag\("flagname"\)](#) "reason"

Functions that have actions but must be preceded by the 'call' action as they are really functions and must be on a line of their own (not on the end of an if statement)

```
call forward\_cc("new@email.address")
call replace("header_name","wildcard_match_pattern","replacement_pattern")
call report("manger@email.address","subject of message")
```

Builtin Functions

```
call\_add\_header("Header: header information")
allmod()
exists("header")
head\_len("header")
isbase64()
isbinary() )
isencodedhtml()
\(isencodedtext()
isencodedurl()
isflag("flag-name")
ishtml()
isimage()
isin("header","string-not-case-sensitive")
lines()>3)
match("header","wildcard")
matchall("header","wildcardlist")
matchone("header","wildcardlist")
rexp("header","regular-expression")
size()
call\_spamdetect(n,"reason")
call\_spawn("d:/surge/filter.exe $FILE$")
```

New Functions

```
time_hour() - returns the 'hours' 0-23, useful for rules that apply at different times of day
time_min() - returns the minutes
isimage() - True if message contains an image
isjpg() - True if message contains a jpeg image
ispdf() - True if message contains a pdf file
image_size() - Approx size of image in bytes
nimage() - Approx number of images found in message
islocal() - Message is to a local user not an outgoing message
isloggedin() - Message is from a logged in local user
is_dayofweek("monday,tuesday") - True on those days of the week.
```

Notes

The "header" parameter can be any normal header, such as "Subject", "From" or "To". However, there are some additional pseudo-headers that can also be used as parameters in any function which takes a "header" parameter:

```
"head": refers to the entire message header.
"body": refers only to the message body (after any necessary decoding)
"urls": refers to any urls found in the body
```

Function Descriptions

call add_header("Header: header information")

Used to add a header to a message. eg

```
if (isin("x-spamdetect","*****") then
call add_header("X-MailScanner-SpamCheck: LEVEL=*****")
```

```
end if
```

NOTE: This will cause bounces if used in local.rul or simple.rul, it can only be used in mfilter.rul
Requires Version 3.8 or later.

allmod("header")

This returns true if all the newsgroups in the specified header are moderated.

exists("header")

This is true if the header exists in the message and is non zero in length, eg: if (exists("supersedes")) then reject "We don't like supersedes headers"

head_len("header")

Returns the length of the named header, e.g.

```
if (head_len("date")>60) bounce "Naughty message"
```

isbase64()

This is true if the message appears to contain base64 binary encoded data.

isbinary()

This is true if the message has binary data either base64 encoding or uuencoded data.

isencodedhtml()

This is true if the message appears to contain MIME or uuencoded HTML instead of plain text data.

isencodedtext()

This is true if the message appears to contain MIME or uuencoded text data. This will always be true if isencodedhtml() returns true.

isencodedurl()

This is true if the message appears to contain an uuencoded URL reference.

isflag("flag-name")

Used to check whether a flag variable has been defined as true. This can be done with the setflag("flag-name") action, e.g.

```
if (size()>100000) setflag("bigitem")
if (isimage()) setflag("bigitem")
if (isflag("bigitem")) reject "It was a big item or had a picture in it"
```

ishtml()

This is true if the message appears to contain HTML instead of plain text data.

isimage()

This is true if the message appears to contain a picture (either MIME or uuencoded)

isin("header","string-not-case-sensitive")

This is a simple 'content' searching function if the named header contains the string (a non case sensitive match is used) eg:

```
if (isin("Subject","Free"))
reject "Probably a spammer selling something"
```

This would reject a message containing a subject of "Get your Free pictures here" it would also reject a message containing a subject of "Is there any real freedom in the world?" so it's probably not a good rule :-)

lines()

This returns the number of lines in the message.

match("header","wildcard")

This function applies a simple wild card matching algorithm as is typically used to match file names, eg:

```
match("From","*@netwin.co.nz")
```

would match against a message from that domain.

matchall("header","wildcardlist")

Used for matching a single wild card against a header which contains a list of values, like Newsgroups:, Path: etc..., The match is TRUE only if all entries in the list match, eg:

```
if (matchall("Newsgroups","news.filters.*")) accept "It is only in the filters list so we will accept it"
```

matchone("header","wildcardlist")

Identical to the above function but returns 'TRUE' if any match occurs.

rexp("header","regular-expression") This function searches the named header for a regular expression, the matching is not case sensitive, use **rexp_case()** for a case sensitive version.

rexp_fast(spamdetect_score,"regular expression ","comment for spam header")

This is just like rexp, but it does the search more efficiently, the first 2 characters of regular expression must be plain ascii (not a regular expression) if it's found in the body of the message then the score is added to the spam_detect header

size()

Returns the size in bytes of the current message can be used with > and < operators.

call spamdetect(n,"reason")

This function can be used to mark a message as possible spam, the 'n' is a (floating-point) number and each time this function is called for a message the total is increased, then finally a header is added to the message;

X-SpamDetect: <stars>: <score> <reason1> [reason2 [reason3 ...]]

<stars> is a string of n stars, where n is the total score (capped at 20)

<score> is the total spam score

The idea is that users can then set their mail clients to filter messages based on this pseudo header. For instance, filtering any message with "*****" in its X-SpamDetect header will throw out any message with a score of 6 or more.

Please note that "local.rul" should be used for adding scoring for ASPAM not mfilter.rul.

call spawn("program.exe \$FILE\$")

This function runs a program on each message the \$FILE\$ macro is replaced by a temporary file name containing the actual mail message. The return value of the program (return n; in main() function) is returned by this 'spawn' function, so it can be used to filter the message or allow it to continue. eg:

```
if (spawn("d:/path/xfilter.exe $FILE$")) reject "That was spam according to xfilter"
```

NOTE: The mfilter is only passed the first 14k of each message, and so the spawned program also only gets the first 14k not the entire message.

Actions

accept "reason"

Accepts the current article reporting the "reason" specified in the log files.

clearflag("flag-name")

Used to set the specified flag variable to the false state.

forward "remote@address.com"

Forwards the message to the specified address and terminates processing.

call forward_cc("new@email.address")

Sends the current message to this new Email address in addition to any existing destination users.

reject "reason" (or bounce "reason")

Rejects the current article reporting the "reason" specified in the log files and to the user

call replace("header_name","wildcard_match_pattern","replacement_pattern")

If the named header matches the 'wildcard_match_pattern' then the replacement pattern is applied, e.g.
 replace("from","*.*.domain.name","BOB_%1@%2.other.name")

Subject: "joe@this.domain.name"

Would be translated to:

Subject: "BOB_joe@this.other.name"

call report("manger@email.address","subject of message")

Sends an Email, including the top part of the offending message, to the specified person, with the specified subject. This is intended when you want to be alerted to something but don't want to simply forward the message itself which may be 'confusing' as it would look like the message had been sent to the manager directly.

setflag("flag-name")

Used to set the specified flag variable to the true state.

Regular Expression Syntax - In Brief

Please note you need to escape spaces in this implementation.

eg:

sweepstake lottery / international program

sweepstake lottery/ international program

sweepstake lottery /international program

So what you want is this. Just put slashes in front of the spaces.

sweepstake lottery(/ | /|)international program

if (rexp("subject","sweepstake lottery(\ ^ | |/\^)international program")) bounce "a"

\s = white space

\S = not white space

\d = digit

\D = not digit

\b = word boundary

\B = not word boundary

\x00 = Hex character

. (period) represents any one character.

[] (brackets) contain a set of characters from which a match can be made. It corresponds to one character in the search string.

**** (backslash) is an escape character which means that the next character will not have a special meaning.

***** (asterisk) is a multiplier. It will match zero or more of the previous character. (Note: it is not a wildcard character as in file names.)

? (question mark) is a multiplier. It will match zero or one of the previous character. (Note: it is not a wildcard character as in file names.)

+ (plus) is a multiplier. It will match one or more of the previous character.

{} (squiggly brackets) contain a number which specifies an exact number of the previous character, or range {2,3}

[^] (brackets containing caret and other characters) means any characters except the character(s) after the caret symbol in the brackets.

^ (caret) is the start of the line.

\$ (dollar) is the end of the line.

(Note the following < > (begin and end word) are not implemented, use \b instead)

[:alpha:] represents any alphabetic letter.

[:digit:] represents any single-digit number.

[:blank:] represents a space or tab.

Lookahead operator

Free(!dom|bsd) matches freesex but not freedom or FreeBSD

OR operator

| (pipe) is OR. It requires that the joined expressions have parentheses around them.

Examples:

e.a matches eta, eda, e1a, but not Eta

[eE].a matches eta and Eta

E.*a matches Eudora, Etcetera, Ea

ho+p matches hop, hoop, hoooop, but not hp

etc\. matches etc. but not etc

Example rule file:

```
$sex = "fuck|xxx|sex"
$free = "free(!dom|bsd|nix|serve)"
$pics = "pi[cx]"
$free_pictures = $free + $pics
$bad_guys = + " |freepictures|jus.?.?.doi.?.?.to|great\.site|webbinaries" \
+ " |yad.?.?.?.ion.?.?.org|freehidden|joy.?.?.to.?.?.al|from.?.behind" \
+ " |love(youhon|ergirl|chatting|stofuck)|forever\.yours|@ju.?.?.sex|town\.girl|beachbums" \i

# Do some processing which is specific to individual recipients
recipients
    if (isin("recipient","manager@this.domain")) accept "Always accept for me so spammers can talk to me"
    if (isin("recipient","sales@your.domain")) then
        if (isin("subject","order")) then
            # Make a Duplicate of sale order
            call forward_cc("sales\_copy@your.domain")
        end if
    end if
end recipients

# Check for some known spammers and naughty subjects
if (rexp(subject,$free_pictures)) bounce "No emails about free pictures"
if (rexp(from,$bad_guys)) bounce "No emails from black listed people thanks"

# Strip local node names from from addresses:
call replace("From","*.*.parts.co.nz","%l@parts.co.nz")

accept "Great, we liked the message"
```

Example 2:

We want to block any message that has been found in SURBL database. We will use the exists function to check if that header exists.

```
if (exists("X-Surbl")) reject "Your SPAM is not wanted here."
```

You can easily change that to drop the message silently if you prefer

```
if (exists("X-Surbl")) drop "SURBL SPAM is not wanted here."
```

(The reason will be logged so still important to put there)

Example 3:

We want to block any message with no subject header. SurgeMail adds a subject header if it is missing so we have to match on the text that SurgeMail adds.

```
if (isin("Subject","(No subject header)")) bounce "No Subject header"
```

Example 4:

We want to block any message with an empty subject header.

```
if (head_len("Subject")<1) bounce "Empty Subject header"
```

Example 5:

I have a user fred in one of my local domains localdomain.com I only want him to be able to send to other users at localdomain.com and not to any other domains.

```
recipients
if (isin("from","fred@localdomain.com")) then
    if (!isin("recipient", "localdomain.com")) bounce "Sorry you can only send to localdomain.com"
end if
```

Friendly Relations System

The Friendly Relations System or "Friends" is a spam prevention system that will check incoming mail based upon a list of known email addresses. The Friends system can operate in the following modes:

- [Kids safe mode](#) - Only accept mail from known friends
- [Keep track of Friends but don't block anything or request confirmation](#).
- [Request confirmation from sender](#)
- [SmiteSpam mode](#) - Request confirmation if SmiteSpam gives it a rating bigger than selected.
- Allow all mail through - Friends system disabled

Kids Safe mode

This is the highest level of safety. Mail is only allowed through if the inbound email address is in the list of known friends. If this is not the case the inbound message is bounced with a notification that this person is not accepting mail from unknown senders.

This safe mode message can be customised at the server level by creating a text file "kids-safe.eml" in the SurgeMail directory.

Keep track of Friends but don't block anything or request confirmation

This mode is most similar to ["Request confirmation mode"](#), the difference is that this mode does not hold messages from unknown senders and it does not send a message to them requesting confirmation. So it simply applies the friends list, accepting known friends and leaving unknown ones for another feature/filter to deal with.

Request confirmation mode

In this mode if a message is received from an unknown user an email is sent to this person requesting them to respond to confirm that they are human. If a response is received the original message will be delivered. On a regular basis a status report message is sent to the user providing information of the friends system and any mail pending delivery.

The Friends system may be temporarily disabled if an urgent message is expected from an unknown address using the "Un-Block" button. This will effectively turn either the "Request confirmation mode" or the "Kids safe mode" into "Allow all mail through" for a period of three hours.

SmiteSpam mode

In this mode if a message is recieved from an unknown user and it gets a SmiteSpam rating of more than selected then a request confirmation message is sent as per the 'request confirmation mode' above.

Configuration:

The friends system must be enabled at server level ([g_friends_only](#)). If enabled at server level, the following settings can be set by the user:

Message

The message sent to users to request confirmation that they are human.

Exceptions

A set of rules defining email that is exempt from the friends email filtering system.

Pending

A list of pending messages currently awaiting confirmation that the sender is human.

Incoming/Outgoing

List of messages received and sent.

Log

Log of actions performed on the Friends system.

This is a CGI program that collects information that a user has submitted in a web page form, and then emails it to someone, and optionally can CC to another address. This could be used for collecting information, or for help requests or bug reports etc...

Problems?

Please read this page, it should contain everything you need to know about setting up EasyForm, only then if you still cannot figure it out, you should email your problem, explaining exactly how you have it set up to EasyForm-support@netwinsite.com

Free Software

Yes, we do not charge for the use of this software. You may copy and distribute it as you wish. As such, it comes with no guarantee of any sort. If you find any bugs, please report them to EasyForm-support@netwinsite.com as we do like to have bug free programs :-)

Platform	Download file
Windows NT, 2000, XP ... (Intel x86)	easyform.exe
Linux libc6 (Intel x86)	easyform.tar.gz

UNIX Install Notes:

```
wget http://netwinsite.com/ftp/easyform/easyform.tar.gz
gunzip easyform.tar.gz
tar -xvf easyform.tar
./easyform.cgi -install
```

Windows Install Notes:

```
Download http://netwinsite.com/ftp/easyform/easyform.exe
Run the self extracting archive.
```

What you will need to edit by hand.

The installer should create all the files you need to make it work, once it's working (and not before) then tailor the form and settings for your requiremens, here are the files you will need to edit and their default places (on windows when used with SurgeMail, it will work with any web server though, e.g. IIS or apache)

File	Description
c:\surgeemail\scripts/easyform.ini	Main ini file, specify the form fields you want sent in the email message, and the destination address and mail server etc.
c:\easyform/failed.htm	Web page shown if the email cannot be sent or if fields are missing
c:\easyform/sent.htm	Web page shown when email is sent ok
c:\surgeemail\www/easyform.htm	Web page that has the sample form, you can use any html editor to add more fields and make it look pretty.
c:\surgeemail\scripts/easyform.exe	The actual binary.

More problems?

Are you getting 404 or 405 errors when you click on the Submit button? The 404 means that the easyform.exe file is not where you thought it was. You'll need to check that the "action" variable inside the "<form>" tag is pointing at the right file, and that the file is in the right place.

The 405 error means that the post cant be sent to the easyform.exe file, because it either has the wrong permissions set on it (unix only problem so far), or that the web server doesnt think that that particular directory is valid for executable files, in this case you need to find the right directory, and put the easyform files there, or consult your web server documentation to see how you are supposed to have cgi compiled script files run.

Once you have done this, the form would contain the variables "Name:". "Email:", "Interests:". These must be specified in the ini file if you wish them to be included in the email.

Ini file settings

There must be no spaces before an ini file setting, one space after it, and the value for the setting goes directly after that space.

Setting	Example	Description
---------	---------	-------------

host	127.0.0.1	Use to specify smtp server, e.g. "smtp 127.0.0.1"
store_fields	name,email,question	Specify the form fields you wish to send in your email
from_field	email	Use this field as the from header in the message
bounce	you@your.domain	This must be your own email address, this is the bounce envelope used when sending the email to you.
to	you@your.domain	The destination address for the email.
cc	friend@your.domain	This address will also receive a copy of the message
subject	Message from web	This defines the subject of the email message.
work	c:\easyform	Specify a path where easyform can write it's log file and any working files it needs.

SurgeNews News Server

Warning this product is no longer being developed, you are welcome to purchase it and use it 'as is'.

Key Features

- New **high performance fully threaded** design.
- **Millions of items.** Database capable of handling millions of items per group, and storing a years news on a single server without suffering degradation as the quantity of data increases.
- **Full feed** - share a full news feed over several serves. Use anything from 1 to 20 servers to spread the load over the most cost effective hardware and cope with any sized user load. ([more info](#))
- **Pull feed** - where bodies are only fetched when messages are read by real users saving huge quantities of bandwidth. Pull from multiple upstream sites! ([more info](#))
- **Built in** efficient clean and tailorable **WebNews interface**. Includes automatic image thumbnail generator and reconstructs multi part items. ([more info](#))
- **Charge/Control users the way you want to.** Control user limits by ip or user, and by day or month usage limits, or by bandwidth. Widearea user limits so limits can apply over a farm of news servers.
- **Proxy mode.** Use SurgeNews/Proxy to front end your system and connect users to multiple back end news servers each serving a subset of news, thus spreading the load between multiple cheap systems rather than trying to build one enormous news server. Proxy mode also includes caching to decrease back end server load. ([more info](#)) (Note back end's can also be split evenly instead of by group ([more info](#)))
- **Spam Proof.** Spammers routinely harvest your users email addresses from newsgroups. SurgeNews includes a unique technology to replace email addresses in headers with ever changing addresses. It then acts as a mail server firewall and only allows responses to the actual news messages to go through to the real email addresses (and only for a fixed time period) ([more info](#))
- **Simple to install.** You can be reading news in minutes and the web management interface will give you full control of all the features - no tricky config files to struggle with!.
- **NOTE: NOT A DNEWS UPGRADE** - this is a separate product, it should not be installed over an existing dnews installation! [Guide to choosing dnews/surgenews.](#)



Download a free 30 day trial now. [Download Now!](#)

Upgrading from DNews to SurgeNews

- **SurgeNews is a newer product**, it has the following advantages.
 - It's fully threaded.
 - It can cope with larger news groups and larger spool areas.
 - It supports a 'proxy' mode for sharing load/groups between several servers.
 - It can do multiple channel header based sucking.
 - It has a built in web admin interface.
 - It has a built in webnews interface.
 - It's easier to install and use.
- On the other hand
 - It doesn't have all the features of dnews.
 - It is not compatible with dnews config settings or dnews database files (you can't trivially upgrade/downgrade)

- We recommend people use SurgeNews for 'new' installations or when upgrading hardware, but generally if dnews is working fine for you, then upgrading would probably be a lot of effort/risk with no obvious benefit.

Menu

Auth Modules

- [About Auth Modules](#)
- [NWAAuth](#)
- [UNIXAuth](#)
- [NTAuth](#)
- [LDAPAuth](#)
- [MySQLAuth](#)
- [MultiAuth](#)
- [OracleAuth](#)
- [PAMAuth](#)
- [RadiusAuth](#)
- [DNAuth](#)
- [ODBCAuth](#)
- [TCPAuth](#)
- [POPAuth](#)
- [HTTPAuth](#)

Auth Protocol

- [Simple Protocol](#)
- [Extended Protocol](#)
- [Module Table](#)

Installation and Setup

- [Setting up SurgeMail](#)
- [Setting up DNews](#)
- [Testing Module](#)

Download Module

Authentication Modules

Authentication modules are used by various NetWin Ltd products which include:

- [SurgeMail](#) (and DMail)
- [DNews](#)
- [DBabble](#)

These products use the external module to determine what accounts are valid and they can be used to store other useful information about the user like disk quota.

The modules that are displayed on this page are ones that we currently provide. Since all the modules follow the same '[Authentication Protocol](#)', you can write your own modules to interface with the database that you wish to use. You can also contact NetWin Ltd if you wish us to write the module for you. If you have any questions or need to know more about any aspect of authentication modules please Email:

support@netwinsite.com

Contents

- [Overview](#)
- [Currently Available Modules](#)
 - [NWAAuth](#)
 - [UNIXAuth](#)
 - [NTAuth](#)
 - [LDAPAuth](#)
 - [MySQLAuth](#)
 - [MultiAuth](#)
 - [OracleAuth](#)
 - [PAMAuth](#)
 - [RadiusAuth](#)
 - [DNAuth](#)
 - [ODBCAuth](#)
 - [TCPAuth](#)
 - [POPAuth](#)
 - [HTTPAuth](#)

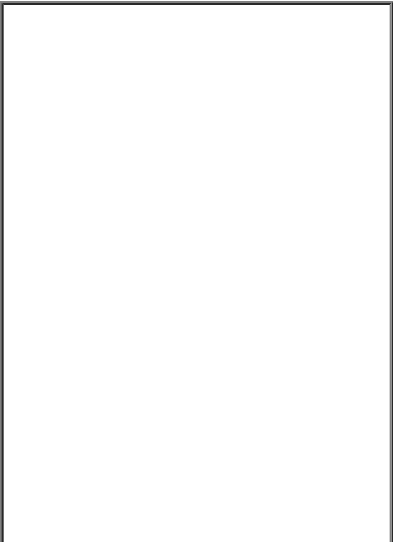
Overview

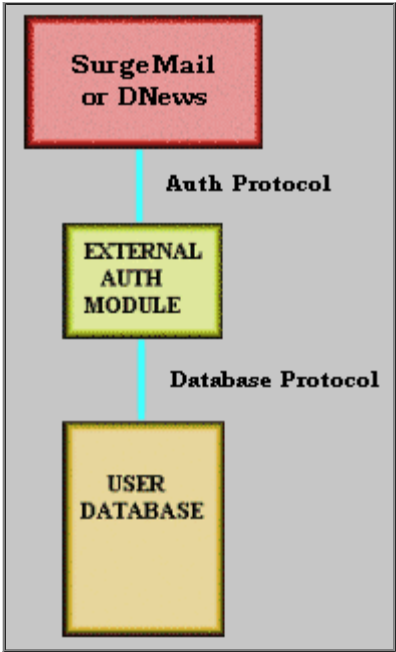
There are many different types of external authentication modules, some do have their limitations due to the interface to the database being used. Each is explained in the sections that follow.

The diagram to the right displays the overall view of where the external authentication modules fits compared with the main application (like SurgeMail and DNews) and the database where the user information is stored.

The protocol that the main application uses to talk to the external authentication module is defined in the '[Authentication Protocol](#)' section of this manual. Not all modules will support all protocols due to limitations on accessing data in the main database.

This manual will not deal with any information about how to interface the external auth module with the final target database. There is plenty of





information on the internet which would help you in this.

The biggest advantage of setting up the external module like this, is that if we do not have a module to interface with your target database you can write your own module following the '[Authentication Protocol](#)'.

Modules Currently Available

There are many different types of external authentication modules available. Each module is describe in the following table. If you click on the name of the module it will take you to a page giving more deatils about that module.

Link for Details	Overview
NWAuth	<p>This external authentication module comes in all distribution sets. The source is provided on all platforms and for Windows and most UNIX based platforms it is pre-compiled, as nwauth.exe or nwauth.</p> <p>This module is the default module that most NetWin Ltd products use. This uses a simple text file were all the user information is stored. All passwords are encoded using 'crypt'.</p> <p>This is our recommendation.</p>
UNIXAuth	<p>UNIXAuth should ONLY be used if you have an existing mail server whose email accounts are in fact UNIX user accounts, it should not be used otherwise UNLESS you want to give your email users a shell account on your UNIX system.</p> <p>UNIXAuth will only work if run as root. You can force it to run as root even if the application calling it isn't.</p> <p>chown root:root unixauth chmod 6775 unixauth</p>
	NTAuth is version simular to that of

NTAuth	<p>UNIXAuth except that the module only works on Windows systems.</p> <p>This module should ONLY be used if you have an existing mail server whose email accounts are in fact Window user accounts.</p>
LDAPAuth	<p>This module allows your user information to be stored in an LDAP database. LDAPAuth should work with any v2 or v3 compliant LDAP server.</p> <p>Recommended LDAP servers:</p> <p>SurgeLDAP (By Netwin Ltd)</p> <p>OpenLDAP</p>
MySQLAuth	<p>This module interfaces with a MySQL Database.</p> <p>Recommended MySQL servers:</p> <p>MySQL</p>
MultiAuth	<p>This module allows you to authenticate with several modules simultaneously based on wild card matching and other rules.</p> <p>This allows you to setup 1 (or more) domains each using a different database. For example you could setup 2 domains where each domain has seperate LDAP databases. So you setup two LDAPAuth within MultiAuth.</p>
OracleAuth	<p>This is our module to talk to an Oracle user database.</p>
PAMAuth	<p>This is the module to talk to the a Linux PAM module.</p>
RadiusAuth	<p>This is the module to talk to the a Linux Radius module.</p>
DNAuth	<p>This is a modified verison of NWAAuth which can check and lookup users from a DNews users.dat file.</p>
ODBCAuth	<p>Our authentication module for talking to an ODBC Driver for a Database (e.g. MS Access, MS SQL Server, ORACLE)</p> <p>This is only available for Windows NT/2000/XP systems.</p>
TCPAuth	<p>TCPAuth is a TCPIP client+server module that will take any other module as its backend. Allows easy authentication across boxes and across platforms.</p>
POPAuth	<p>POPAuth is a POP proxy authentication module. Allows one or more POP servers to be used for authenticating users.</p>

HTTPAuth	HTTPAuth is an HTTP proxy authentication module.
--------------------------	--

CentiPaid System

(available in 1.4c and greater)

The CentiPaid System is a system that allows you as the admininstrator or selected users to charge senders for incoming email. For a good description of the system and how it works see <http://www.centipaid.com/>.

Centipaid.com has kindly created their own step by step SurgeMail configuration instructions, they can be found here: <http://www.centipaid.com/en/support/surgemail.html>

How does it work?

When your account recieves an incoming email, it is checked for a payment reciept, if there is no reciept then a message is bounced to the sender instructing them that payment is required to deliver their message. The bounce contains a link to click in order to pay.

They pay, and an email is sent from the CentiPaid server to SurgeMail informing it of the payment, SurgeMail then verifies the payment reciept via the CentiPaid authentication server and releases the original message to the account.

The contents of the bounce message are customisable, the default message is hard-coded. Users can define their own personal message, or you can define one in a file called centipaid.msg (or centipaid.eml) in the SurgeMail installation directory. It will use that unless the user has defined one.

Configuration

Global settings, [g_centipaid](#), eg.

```
g_centipaid "pay001.centipaid.com:2021"
```

The above setting instructs SurgeMail to use the address and port "pay001.centipaid.com:2021" for verification of payments. If the CentiPaid server is down then messages are assumed to be paid and are delivered without verification.

Domain settings, [centipaid](#), [user_centipaid](#), eg.

```
centipaid match="*" acct="AEF001" pass="adonis" https="http://pay.centipaid.com/index.php" lang="en"
amount="0.005" enabled="TRUE" friends="FALSE" smite="0"
```

The above setting tells SurgeMail that in this domain accounts matching 'match' may enable and use centipaid.

Field	Meaning
match	Wildcard, if this matches an account name then that account can have CentiPaid enabled.
acct	The merchant account number.
pass	The merchant / authentication server password.
https	The link to the page that accepts payment from the user.
lang	The language to display the above page in.
amount	The amount to charge per email.
enabled	If TRUE then CentiPaid is enabled for all matching accounts, otherwise the user may enabled it (depending on value of user_centipaid)
friends	If TRUE then senders marked as Friends will be charged, otherwise they will be allowed.
smite	SmiteCRC level, if an email receives more than this it is charged. A value of 0 charges all emails.

```
user_centipaid "acct,pass,https,lang,amount,enabled,friends,smite"
```

This setting specifies the various CentiPaid configuration options already discussed that the user can customise for themselves. For example allowing users to customise acct and pass means they can themselves become a Merchant.

SurgePlus - Schedule, File and Photo Sharing

User Help

For user help on SurgePlus see the SurgePlus page on your SurgeMail server. e.g. <http://127.0.0.1:7080/surgeplus/> or refer to our page at <http://netwinsite.com/surgeplus/>

Administrator Help

The SurgePlus client is free.

Controlling access to SurgePlus

You can disable SurgePlus with the `g_disable_surgeplus` setting. SurgePlus uses the same user database as SurgeMail uses for email. By default, all your email users are able to use SurgePlus. You can selectively allow access to SurgePlus using the domain `user_access_default` setting or for individual users using their authentication database `user_access` field.

For example, if you want to allow only a single domain to use SurgePlus, then you would set your `g_user_access_default` setting to "all,!surgeplus", the `user_access_default` setting for the particular domain you want to allow to "all". If there is a single user within a second domain that you want to allow SurgePlus access to, you could give their `user_access` database field a value of "all".

Downloading and customizing the SurgePlus client

SurgePlus is automatically downloaded from netwinsite.com to your SurgeMail server to be made available for download by your users. Use the `g_disable_surgeplus_updates` setting to stop this. SurgeMail will automatically configure it's local copy of the client download to connect back to your server. To download an automatically configured SurgePlus client from your server make sure you use a valid SurgeMail domain name in the URL. For example "http://your.domain.name:7026/surgeplus/". In this case the SurgePlus client will be automatically configured to use your.domain.name as the server to connect to.

You can further customize the SurgePlus client (for example changing default preferences or images). See the 'Default Preferences', 'Appearance', 'Email Template', and 'Web Template' sections of your SurgePlus administrator web interface.

Mirroring Notes

If you are using a mirror server, then all SurgePlus data is mirrored to your mirror server. Your users are able to connect to either server to use SurgePlus.

Menu

Auth Modules

- [About Auth Modules](#)
- [NWAAuth](#)
- [UNIXAuth](#)
- [NTAuth](#)
- [LDAPAuth](#)
- [MySQLAuth](#)
- [MultiAuth](#)
- [OracleAuth](#)
- [PAMAuth](#)
- [RadiusAuth](#)
- [DNAuth](#)
- [ODBCAuth](#)
- [TCPAuth](#)
- [POPAuth](#)
- [HTTPAuth](#)

Auth Protocol

- [Simple Protocol](#)
- [Extended Protocol](#)
- [Module Table](#)

Installation and Setup

- [Setting up SurgeMail](#)
- [Setting up DNews](#)
- [Testing Module](#)

Download Module

Authentication Modules

Authentication modules are used by various NetWin Ltd products which include:

- [SurgeMail](#) (and DMail)
- [DNews](#)
- [DBabble](#)

These products use the external module to determine what accounts are valid and they can be used to store other useful information about the user like disk quota.

The modules that are displayed on this page are ones that we currently provide. Since all the modules follow the same '[Authentication Protocol](#)', you can write your own modules to interface with the database that you wish to use. You can also contact NetWin Ltd if you wish us to write the module for you. If you have any questions or need to know more about any aspect of authentication modules please Email:

support@netwinsite.com

Contents

- [Overview](#)
- [Currently Available Modules](#)
 - [NWAAuth](#)
 - [UNIXAuth](#)
 - [NTAuth](#)
 - [LDAPAuth](#)
 - [MySQLAuth](#)
 - [MultiAuth](#)
 - [OracleAuth](#)
 - [PAMAuth](#)
 - [RadiusAuth](#)
 - [DNAuth](#)
 - [ODBCAuth](#)
 - [TCPAuth](#)
 - [POPAuth](#)
 - [HTTPAuth](#)

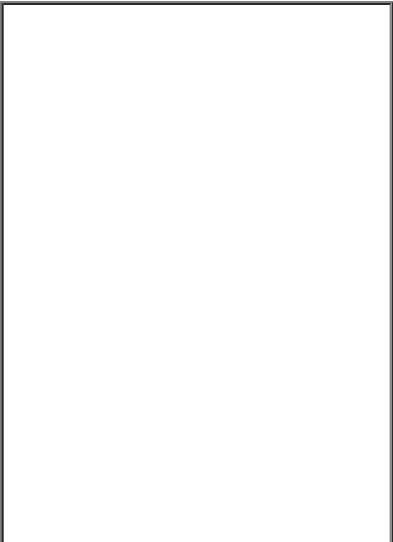
Overview

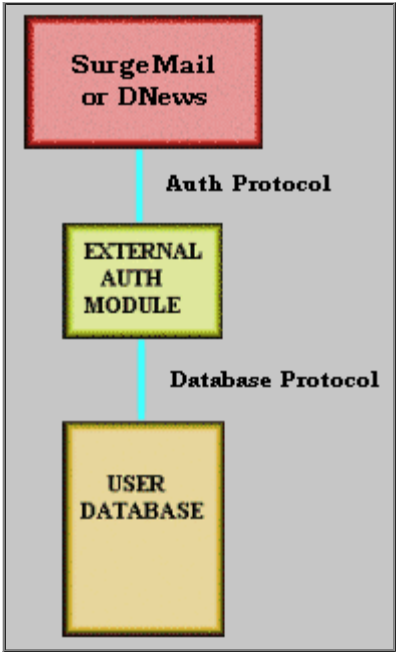
There are many different types of external authentication modules, some do have their limitations due to the interface to the database being used. Each is explained in the sections that follow.

The diagram to the right displays the overall view of where the external authentication modules fits compared with the main application (like SurgeMail and DNews) and the database where the user information is stored.

The protocol that the main application uses to talk to the external authentication module is defined in the '[Authentication Protocol](#)' section of this manual. Not all modules will support all protocols due to limitations on accessing data in the main database.

This manual will not deal with any information about how to interface the external auth module with the final target database. There is plenty of





information on the internet which would help you in this.

The biggest advantage of setting up the external module like this, is that if we do not have a module to interface with your target database you can write your own module following the '[Authentication Protocol](#)'.

Modules Currently Available

There are many different types of external authentication modules available. Each module is describe in the following table. If you click on the name of the module it will take you to a page giving more deatils about that module.

Link for Details	Overview
NWAuth	<p>This external authentication module comes in all distribution sets. The source is provided on all platforms and for Windows and most UNIX based platforms it is pre-compiled, as nwauth.exe or nwauth.</p> <p>This module is the default module that most NetWin Ltd products use. This uses a simple text file were all the user information is stored. All passwords are encoded using 'crypt'.</p> <p>This is our recommendation.</p>
UNIXAuth	<p>UNIXAuth should ONLY be used if you have an existing mail server whose email accounts are in fact UNIX user accounts, it should not be used otherwise UNLESS you want to give your email users a shell account on your UNIX system.</p> <p>UNIXAuth will only work if run as root. You can force it to run as root even if the application calling it isn't.</p> <pre>chown root:root unixauth chmod 6775 unixauth</pre>
	NTAuth is version similar to that of

NTAuth	<p>UNIXAuth except that the module only works on Windows systems.</p> <p>This module should ONLY be used if you have an existing mail server whose email accounts are in fact Window user accounts.</p>
LDAPAuth	<p>This module allows your user information to be stored in an LDAP database. LDAPAuth should work with any v2 or v3 compliant LDAP server.</p> <p>Recommended LDAP servers:</p> <p>SurgeLDAP (By Netwin Ltd) OpenLDAP</p>
MySQLAuth	<p>This module interfaces with a MySQL Database.</p> <p>Recommended MySQL servers:</p> <p>MySQL</p>
MultiAuth	<p>This module allows you to authenticate with several modules simultaneously based on wild card matching and other rules.</p> <p>This allows you to setup 1 (or more) domains each using a different database. For example you could setup 2 domains where each domain has seperate LDAP databases. So you setup two LDAPAuth within MultiAuth.</p>
OracleAuth	<p>This is our module to talk to an Oracle user database.</p>
PAMAuth	<p>This is the module to talk to the a Linux PAM module.</p>
RadiusAuth	<p>This is the module to talk to the a Linux Radius module.</p>
DNAuth	<p>This is a modified verison of NWAAuth which can check and lookup users from a DNews users.dat file.</p>
ODBCAuth	<p>Our authentication module for talking to an ODBC Driver for a Database (e.g. MS Access, MS SQL Server, ORACLE)</p> <p>This is only available for Windows NT/2000/XP systems.</p>
TCPAuth	<p>TCPAuth is a TCPIP client+server module that will take any other module as its backend. Allows easy authentication across boxes and across platforms.</p>
POPAuth	<p>POPAuth is a POP proxy authentication module. Allows one or more POP servers to be used for authenticating users.</p>

HTTPAuth	HTTPAuth is an HTTP proxy authentication module.
--------------------------	--

Customising surgemail web templates

SurgeMail web template files (for NetAuth and SurgePlus) can be customized in a similar way to WebMail template files to allow you to change the contents or style of any of the web interface pages.

SurgeMail web templates use a similar (but not identical) template parsing system to that of WebMail (see [WebMail template customization documentation](#))

Contents

- [Introduction](#)
- [SurgeMail Style Notes](#)
- [Variables](#)
- [Comments](#)
- [Including pages within other pages](#)
- [Functions](#)
- [Conditional Statements \(if, else, endif, etc\)](#)
- [Repeating sections](#)
- [Variables available in all pages](#)
- [Caching](#)
- [Miscellaneous Notes](#)
- [Complete Syntax Reference](#)
- [Complete Function Reference](#)

Introduction

Each web page a user views when using SurgeMail is generated from a template page that is stored in the **web** directory within the SurgeMail installation directory. To find which source file a page you are viewing is generated from, view the page source. There will be a comment at the start with the source filename. (you can prevent source names appearing in output pages using the `g_web_hide_source_names` setting). Each template page is made up of standard html text, with the addition of a variable reference scheme to allow the content of the pages to change depending on what the user does.

SurgeMail Style Notes

If you are just wanting to change the overall look and feel of a SurgeMail web interface then you may just want to change some of the standard css values specified in [surgemail.css](#) in the SurgeMail web directory.

All SurgeMail admin, SurgePlus, and NetAuth pages include [style.htm](#) at the top of the output pages. This page itself includes references to the default [surgemail.css](#) and [surgemail.js](#) files.

Variables

Each variable is referenced by surrounding the name of the variable with double bars (||). For example, on the SurgePlus page that appears at the top of all pages the user sees while using SurgePlus (`surgeplus_web_top.htm`) something similar to the following text appears:

```
||user_field_full_name|| (||user_email||)
```

Here, the font tags are standard html font tags and the text `||user_field_full_name||` gets replaced by the users full name, and the text `||user_email||` gets replaced by the users email address. For example, in the output html page, the text may appear like

```
John Smith (john@netwinsite.com)
```

Notes:

To avoid problems with variable references being confused with JavaScript or symbols, it is necessary to use the

JavaScript or symbol with a space character on either side of it.

To actually display two bars in a web page, use `||`. Most of the variables that are referenced in existing template pages are only available on that particular page. However, there are a number of variables that are defined for all template pages.

Comments

Any line starting with a `#` character is a comment line and is ignored when generating the output web page. It is preferable to use this system of commenting rather than the html comment system, as the `#` template comments do not appear in the output web page, effectively reducing the size of the page that the user downloads. If you require an output page to start with a `#`, use `##` at the start of a template source line and the remainder of the line after the first `#` will be rendered normally.

Including pages within other pages

Often it is useful to include some template pages within other template pages. For example, the page `surgeplus_web_top.htm` appears at the top of all pages a user can see when logged in to the SurgePlus web interface. This is achieved by each of these other pages including the line

```
||include||surgeplus_web_top.htm||
```

at the top of the page. A single page can include multiple other pages and included pages can include other pages within them.

Functions

Often a variable available in a template page is not in a format you want. For example if you have a large number, it is nice to display it with commas separating every third digit. So instead of 5242880 you want the text 5,242,880 to appear. In this case if the variable called `file_size` contained the number 5242880, then you can apply the function called `comma` to this number. This is done using the following syntax:

```
||comma(file_size)||
```

instead of just

```
||file_size||
```

Functions can be applied to the output of other functions and can take multiple parameters. If you need a parameter to a function to be treated as text (rather than the name of a variable) then you must enclose the text in brackets. For example if the variable called `test` has a value of `Hello`, then

```
||trim(test,3)||
```

outputs the text

```
Hel
```

but

```
||trim((test),3)||
```

outputs the text

```
tes
```

Note that the parameter 3 is not enclosed in brackets. This is because if the parameter starts with a digit, it is assumed to be a literal value rather than a variable name.

Here is a more complicated example. (if this is your first reading through this page you may want to skip over it for now)

```
<select name="pref_calendar_day_start_hour">
||select_options(first_defined_of(pref_calendar_day_start_hour,8),(0,...,23),(hour_text))||
</select>
```

`select_options` is the name of a function that takes 3 parameters. The first being the currently selected value, the

second a list of select options and the third parameter is the name of a function to translate the select options into text values.

The first parameter to the `select_options` function in this case is `first_defined_of(pref_calendar_day_start_hour,8)`. This calls the function `first_defined_of`, which means if there is a variable defined called `pref_calendar_day_start_hour` then it uses that, otherwise uses a value of 8.

The second parameter to the `select_options` function takes a comma separated list of select values. The `select_options` function expands the ... to include all numbers between 0 and 23.

The final parameter to `select_options` is the name of another function called `hour_text`. This function takes an integer parameter between 0 and 23 and displays it as a nice string of text identifying an hour. For example 3 pm.

Here is what the final output of this function may look like:

```
<select name="pref_calendar_day_start_hour">
<option value="0">12 am</option>
<option value="1">1 am</option>
<option value="2">2 am</option>
<option value="3">3 am</option>
<option value="4">4 am</option>
<option value="5">5 am</option>
<option value="6">6 am</option>
<option value="7">7 am</option>
<option value="8" selected>8 am</option>
<option value="9">9 am</option>
<option value="10">10 am</option>
<option value="11">11 am</option>
<option value="12">12 pm</option>
<option value="13">1 pm</option>
<option value="14">2 pm</option>
<option value="15">3 pm</option>
<option value="16">4 pm</option>
<option value="17">5 pm</option>
<option value="18">6 pm</option>
<option value="19">7 pm</option>
<option value="20">8 pm</option>
<option value="21">9 pm</option>
<option value="22">10 pm</option>
<option value="23">11 pm</option>
</select>
```

For a complete list of functions available, see [Complete list of template functions](#)

Conditional Statements (if, else, endif, etc)

Often you only want a section of a page to appear under certain conditions. Earlier on we used the following example for variables:

```
||user_field_full_name|| (||user_email||)
```

However, the user's full name may not have been specified. So it is nicer to do something like this instead:

```
||ifdef||user_field_full_name||
user_field_full_name|| (||user_email||)
||else||
user_email||
||endif||
```

`||ifdef||` (meaning if defined) displays all the text between the `||ifdef||` and the `||else||` clause if the variable `user_field_full_name` is defined (and not an empty string). If `user_field_full_name` is not defined, it displays the text between the `||else||` and `||endif||` section. The else section is optional and you can also extend it to use `elseifdef` to display one of more than two options. For example, in the SurgePlus calendar month view, we want the day to appear in green if it is today, black for a standard day, and gray if it is a day from the previous or next month.

```
<a class=||ifdef||is_today||green||elseifdef||is_current_month||black||else||gray||endif|| ...
```

If the variable `is_today` is defined, then it will appear as

```
<a class=green ...
```

Otherwise, if the variable `is_current_month` is defined, then it will appear as

```
<a class=black ...
```

In all other situations it will appear as

```
<a class=gray ...
```

Repeating Sections

Often it is useful to repeat sections of a page multiple times where variables contain different values. For example, when listing the contents of a shared files folder, a section of the page is repeated for each file in the folder. This is done using `||begin_list||` and `||end_list||` pairs. For example

```
<table>
||begin_list||
<tr><td>||file_name||</td><td>||kbytes(file_size)||</td></tr>
||end_list||
</table>
```

So that section of the page between the `begin_list` and `end_list` is repeated for each file in the folder, and the variables `file_name` and `file_size` may have different values each time it is repeated.

The actual text being displayed by `begin_list` and `end_list` will vary depending on the page being displayed. If a page needs to contain more than one repeating list section, then the `begin_list` and `end_list` will have a unique suffix for each section. For example `||begin_list_path||` and `||end_list_path||`

Variables available in all pages

Most variables used by template pages are only available within that particular page. However, some variables are available in all template pages. You can define your own variables to be available in all pages in `style.htm`. Variables available in addition to those defined in `style.htm` include

list_line_num	Available between any <code> begin_list .. end_list </code> pair. This variable contains a count of the list line number, starting with 1 for the first list item.
list_is_first_line	Available between any <code> begin_list .. end_list </code> pair. This variable contains a value of "true" for the first list item and is undefined for later items.
list_line_is_even	Available between any <code> begin_list .. end_list </code> pair. This variable alternates between "true" and undefined on each list item callback, starting with undefined for the first list item
web_base_ref	Available in every page. Used as a prefix to the path for all images, css and js files other pages need to reference. This gets replaced by a unique string dependant on your current SurgeMail version so that SurgeMail can instruct the client browser to permanently cache these files (i.e. the browser should never ask for them again). This is intended to improve performance over default web server setups where the client browser checks if there is a new version of every image in a page each time the browser is restarted. If you upgrade your SurgeMail server or change the value of the <code>g_web_ref_path_extension</code> then clients will request a new copy of the images since they are now referenced in a new directory.

Caching

For performance and caching reasons, any image you want to reference in a template page should be referred to using

```

```

instead of

```

```

The reasons for this is explained below

In order to improve performance for both the server and client we have implemented a caching scheme in SurgeMail for caching of images, .css, and .js files. Normally, a web server will serve these files out to a client browser, which will cache copies of those files. But next time the web browser is started, it will check with the

server if there is a new copy of each image, .css, and .js file required by the page. The server will return a "not-modified" response for each file (rather than giving out the entire file again), but the processing of all these requests still slows down the page loading for the user and puts unnecessary load on the server. So, in SurgeMail we have a special scheme where each image (or .css or .js file) that would normally be referenced by /web/test.gif can instead be referenced as /web/caching-number/test.gif. Here, caching-number is some unique text that changes with each new version of SurgeMail.

When SurgeMail receives a request for a file of this form, it instructs the client to permanently cache the file and to never ask for it again. This could be a problem if the file is modified by you or by us in future versions of SurgeMail, which is why the caching-number text varies from version to version. There is a `g_web_ref_path_extension` SurgeMail setting you can use if you manually change your image files.

Another thing to be aware of is that .css and .js files may need to contain image references. Normally .css and .js files do not have template variables in them replaced. However, if a .js or .css file is referenced using `||web_ref_path||` in the reference name, then SurgeMail will apply template variable replacement on these files so that these files can reference images using `||web_ref_path||` themselves. See /web/surgemail.css for an example where we do this.

If you use your own custom web pages referenced using /web/page_name.htm then normally SurgeMail would not apply template variable replacement to these, so you would not be able to reference image files using /web/||web_ref_path||/image_name.gif. To work around this problem, you can either refer to your custom page using /web/-tpl-/page_name.htm (which means SurgeMail will apply template variable replacement to the page. The "-tpl-" option works only in versions 2.1d-8 and later), or you can refer to image files in your page using /web/-cached-/image_name.gif to force it to be permanently cached. You can add any text you want after the -cached- text if you happen to change your images occasionally. For example /web/-cached-version-1/image_name.gif. You may find the -tpl- option (which forces template variable replacement to occur in the file) useful when using it with .css or .js files which need to contain image references. For example /web/-tpl-/surgemail.css

Miscellaneous Notes

Any template source line ending with a \ character will not have the \ character appear in the output page and additionally, the end of line character will also not appear in the output page. You can use this to space a single complicated line out over several lines in the template source to make it easier to read.

Complete syntax reference

Syntax: `||variable_name||`

Example: `||variable_name||`

Description: Displays the contents of the variable called variable_name

Syntax: `||function_name(parameter1,parameter2,...)||`

Example: `||trim(file_name,7)||`

Description: Applies the specified function to the given parameters. Parameters can either be variables, other function calls, or literal text. If require a parameter to be literal text that does not start with a digit, you must enclose the literal text in brackets. For example `||trim((Hello),3)||` would give **Hel**

Syntax: `||include||file_name||`

Example: `||include||surgeplus_web_top.htm||`

Includes the specified file within the current page.

Syntax: `||ifdef||variable_name||...||endif||`

or: `||ifdef||variable_name||...||else||...||endif||`

or: `||ifdef||variable_name||...||elseifdef||variable_name2||...||else||...||endif||`

or: `||ifdef||variable_name||...||endif(variable_name)||`

Example: `||ifdef||file_name||File name is ||file_name|||endif||`

If the given variable name is defined, displays the given text. The `||else||`, `||elseifdef||` and `||endif||` parts of this syntax are available when using any of the `||if...||` options described below.

Syntax: `||ifdef||variable_name||...||endif(variable_name)||`
or: `||ifdef||variable_name||...||else(variable_name)||...||endif(variable_name)||`

Example: `||ifdef||file_name||File name is ||file_name|||endif(file_name)||`

This is an optional alternative syntax for `||ifdef||`. It is useful in complicated pages where the `ifdef` and `endif` are not near each other so that you can see more easily which `endif` lines up with which `ifdef`. The template parser will generate a template error if the variable name parameter to the `endif` or `else` does not match the variable name used in the matching `ifdef` or `ifndef`.

Syntax: `||ifndef||variable_name||...||endif||`

Example: `||ifndef||file_name||File name is not defined||endif||`

If the given variable name is not defined, displays the given text.

Syntax: `||ifequal||variable_name||value||...||endif||`

Example: `||ifequal||file_name||test.htm||The file is called test.htm||endif||`

If the given variable value is equal to the given value, displays the given text.

Syntax: `||ifnequal||variable_name||value||...||endif||`

Example: `||ifnequal||file_name||test.htm||The file is not called test.htm||endif||`

If the given variable value is not equal to the given value, displays the given text.

Syntax: `||begin_list||...||end_list||`

Example: `||begin_list||file name=||file_name||
||end_list||`

Repeats the text between the `||begin_list||` and `||end_list||` with variables taking on a different value each time.

Authentication Protocol

Various NetWin Ltd products allow use of an External Authentication module to provide for interaction with any type of user database. The external authentication module is run as a sub process and accepts commands from the server on stdin and returns replies on stdout.

The protocol is broken up in two main sections:

Simple Authentication Protocol

All external authentication modules are required to support these commands.

Extended Authentication Protocol

This includes other commands which we recommend your authentication module should implement. These allow for much easier user administration (including adding users) via external utilities.

Our "standard" NWAAuth module also has a number of other commands (mostly only operative from the command line) which are useful for things like, debugging, converting from other systems etc.

Contents

- [Simple Authentication Protocol](#)
 - [Notes on the Definition](#)
- [Extended Authentication Protocol](#)
- [Command prompt commands that should be supported](#)
- [Examples](#)
- [Module list and Supported Commands](#)

Simple Authentication Protocol

The application that is the external authentication module must read commands from standard in and write replies to standard out. **Note:** Not all applications will use all the information that the reply returns.

Every command within the external authentication module can give one of four responses:

- +DATA** A command can return one or more '+data' responses where, the information asked for was successful and it is returning multiple information. The module **MUST** still return '+OK' once all the '+DATA' information has been sent.
 - +OK** This implies that the command was successful.
 - ERR** This is a failed result. Which can mean that the user doesn't exist if a check is performed. Or the data inputted is invalid. It can also provide extra information after the '-ERR'.
- The total response text **SHOULD** be less than 100 bytes.
- DEAD** This response should be returned if for any reason the authentication module is temporarily unable to respond. For example if the user database to which the module connects is down or the module has not been correctly setup.
- The total response text **SHOULD** be less than 100 bytes.

IMPORTANT: Immediately following the +OK, -ERR and -DEAD response tag should come the username exactly as given by the command, this allows applications to verify commands and responses stay in sync. The exception to this rule is a command which does not involve a specific username, for example; search, quit, exit, version, verbose. +DATA is also expected to be followed immediately by the username (or result) but is not used to check things are in sync.

The three basic commands are:

exit

Response: +OK successfull

The exit input tells the external authentication program to shutdown. It should respond with +OK.

check username password [fromIPAddress]

+OK username drop_file_path uid [info]

Response: -ERR username reason
-DEAD username reason

username: The username with optional prefix or suffix to indicate the users domain.
This is generally in the form that the user is to be known by in the user database. This can also be case sensitive.
Some applications will add on '@domain' as a suffix to the username if they have been setup to do so.

password: The users password. This password is case sensitive and the application might have already lowercased the password before it's passed to the external module.

fromIPAddress: The IP address from which the user connected. This is optional, but if given the external module should check against an IPMask. If it does not match then it should return back:
-err ipmask failed

drop_file_path: Full path to the dropfile, including filename, for the user OR the word 'config'.
The keyword config indicates that the servers should work out the user's drop path from the relevant settings it has.

uid: User ID, can be a number or a name or zero, but it must not be left off.
If a number, then UNIX uses that uid number when it checks the ownership of the drop file.
If a name, then DPOP/DSMTTP looks up that name in the UNIX password file and uses the corresponding uid.
If uid is the number 0, then the drop file ownership is not checked or set.

info: Any additional fields that you wish to have. In the layout:
field1_name="value1" field2_name="value2" ...

lookup username

+OK username drop_file_path uid [info]

Response: -ERR username reason
-DEAD username reason

username: The username with optional prefix or suffix to

indicate the users domain.

This is generally in the form that the user is to be known by in the user database. This can also be case sensitive.

Some applications will add on '@domain' as a suffix to the username if they have been setup to do so.

drop_file_path: Full path to the dropfile, including filename, for the user OR the word 'config'.
The keyword config indicates that the servers should work out the users drop path from the relevant settings it has.

uid: User ID, can be a number or a name or zero, but it must not be left off.
If a number, then UNIX uses that uid number when it checks the ownership of the drop file.
If a name, then DPOP/DSMTP looks up that name in the UNIX password file and uses the corresponding uid.
If uid is the number 0, then the drop file ownership is not checked or set.

info: Any additional fields that you wish to have. In the layout:

field1_name="value1" field2_name="value2" ...

Notes on the Definition

1. The authentication module should only ever return one line, in response to any query. The only exception to this is the search command in the extended protocol where a number of +DATA lines are returned.
 2. The returned drop file path must contain the full path and filename of the drop file for that user.
 3. If any directory hashing is required it must be included in the returned drop_file_path string.
 4. The uid field must always be returned.
 5. If the normal drop file path specified in the configuration file is to be used then the drop file path in the response may be replaced by the single word config
 6. The uid is used to set/check user ownership of drop files.
 7. The uid returned may be numeric or an alphanumeric username.
 8. If no uid checking/setting for drop files is to be used then the uid returned can be 0.
 9. Currently the protocol does not support spaces in usernames. If you require this then please contact [DMail Support](#).
 10. The maximum length of any response is 1000 characters in total.
-

Extended Authentication Protocol

The following is a list of extened Authentication command, the following are optional, but if you can set these up you should.

set username [password [info]]

Response: +OK username [message]
-ERR username reason
-DEAD username reason

username: The username with optional prefix or suffix to indicate the user's domain.
This is generally in the form that the user is to be known by in the user database. This can also be case sensitive.
Some applications will add on '@domain' as a suffix to the username if they have been setup to do so.

password: The users password. This password is case sensitive and application might have already lowercased the password before it's passed to the external module.
If the password is given as '(NULL)' then the users info field information is altered only (the password is not changed).

info: Any additional fields that you wish to have. In the layout:

field1_name="value1" field2_name="value2" ...

message: The message is optional,
"user x data updated"

Use this command to ADD or MODIFY a user.

NB: NO warning is given if you are overwriting an existing user.

del username

Response: +OK username [message]
-ERR username reason
-DEAD username reason

username: The username with optional prefix or suffix to indicate the users domain.
This is generally in the form that the user is to be known by in the user database. This can also be case sensitive.
Some applications will add on '@domain' as a suffix to the username if they have been setup to do so.

message: The message is optional,
"+OK Deleted user successfully"

search string [-from <x>] [-max <n>]

Response: +DATA username [info]
+OK [message]
-ERR username reason
-DEAD username reason

search: This is the search on the username which can continue '*' and '?' wildcards. Also some external module might have limits on the minimum number of character that are allowed. (ie: 3)

-from: where -max limits the maximum number of search

-max: results to be returned to n and

-from causes results to be displayed starting at the

x'th match.

- username:

The username with optional prefix or suffix to indicate the users domain.
This is generally in the form that the user is to be known by in the user database. This can also be case sensitive.
Some applications will add on '@domain' as a suffix to the username if they have been setup to do so.
- info:

Any additional fields that you wish to have. In the layout:

field1_name="value1" field2_name="value2" ...
- message:

The message is optional,
"+OK X out of Y results found"

NOTE: SurgeMail will look for the text "out of" in the +OK response, if found it will attempt to locate X and Y and use these values for display purposes. This information is optional and if it requires significant processing to obtain then it's probably best not to include it.

Command prompt commands that should be supported

The external authentication module should not only support all of the commands to be run in interactive mode but also it supports them all run in command line mode, eg:

```
\dmail\nwauth -lookup bob
+OK bob config 0
```

The syntax is generally,

```
NWAuth -command cmdl_param1 cmdl_param2
```

Examples

NWAuth is our example of authentication module code, see [NWAuth Code](#) . You will also find the version of nwauth.c in most of NetWin Ltd applications which supports external authentication.

Set:

```
\dmail\nwauth
set bob password
+OK bob added to database
set fred pass fwd="$USER,bob"
+OK fred added to database
exit
```

Set: command line mode

```
\dmail\nwauth -set bob password
+OK bob added to database
```

Module list and Supported Commands

	Simple Authentication Protocol			Extended Authentication Protocol				Other		
Module	Check	Lookup	Exit	Set	Delete	Search	Help	Quit	Version	Verbose
NWAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	
UnixAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	
NTAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	

LDAPAuth	Y	Y	Y	Y	Y	Y	Y	Y		Y
MySQLAuth	Y	Y	Y	Y	Y	Y	Y	Y		Y
MultiAuth	Y	Y	Y	Y	Y	Y		Y	Y	
OracleAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	
PAMAuth	Y	Y	Y	Y			Y	Y	Y	Y
RadiusAuth	Y	Y*	Y					Y		Y
DNAuth	Y	Y	Y							
ODBCAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	
TCPAuth	Y	Y	Y	Y	Y	Y	Y	Y	Y	
POPAuth	Y	Y	Y							

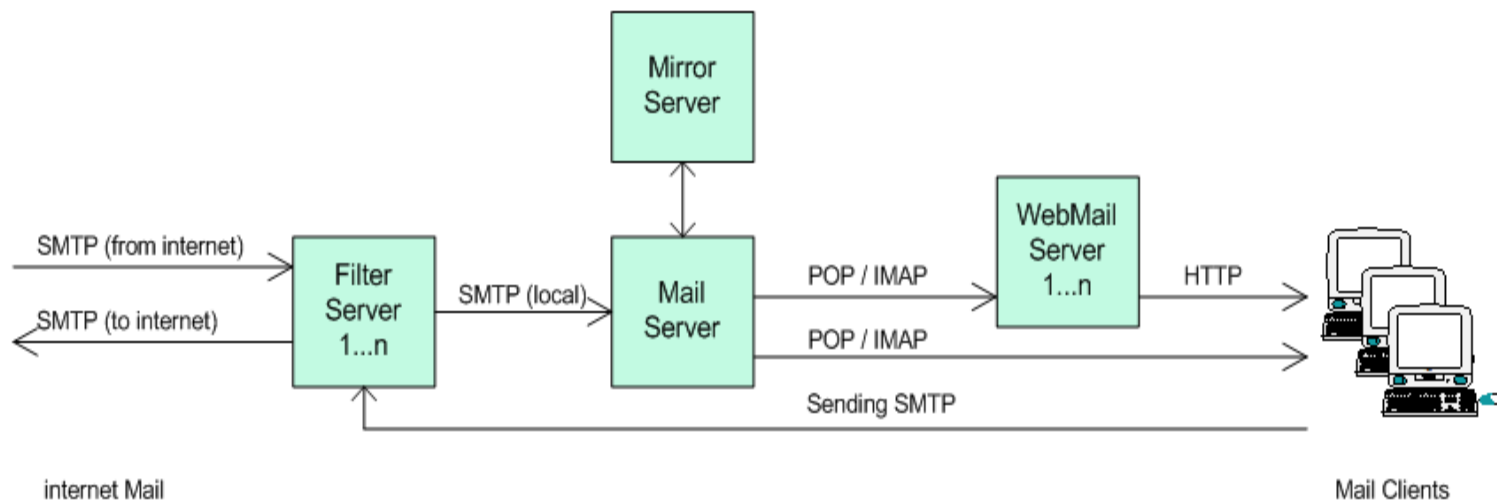
Configuring 'Functionally Split Clusters'

SurgeMail can be functionally split across several servers. The main reason to use this is if your mail load is too large for one server (eg 40000 user+) and / or you have a particularly heavy spam loading or webmail client loading.

You can pick and match what you want to support on each server, but typically you would setup say 2 front end systems for spam and virus filtering. A single mail system to handle storage of local mail including access to this using POP and IMAP. And one or more webmail systems which handle the webmail load and talk to the primary mail server when necessary using IMAP.

This is the most efficient way to implement a high reliability system with a high level of scalability. Dependant on your user needs this allows you to host up to 100,000 users on your primary mail system.

NEVER implement front end filters like this unless you have to, it is always a mistake to have front end servers unless your load leaves you with no other choice!



Other considerations:

- A functionally split architecture can be combined with mirroring. You would simply introduce one more system into the above architecture which is a mirror of the primary mail system. As per mirroring this removes the single point of failure (with associated mail data loss) you would otherwise have if your main mail system were to fail.
- Mail will continue to be accepted by the filter systems if there is a problem with your primary mail system.

Settings you will need to use:

On the 'back end' servers

```
g_relay_allow_ip "ip.of.filter.server*"
g_gateway_allow "ip.of.filter.server"
g_friends_ignore_trusted "true"
g_spf_skip "ip.of.filter.server*"
g_verify_mx_skip "ip.of.filter.server*"
g_spf_share "ip.of.all.other.servers" (this is a list, not a wild card)
g_xfile_allow "*"
g_autologin_pop "true"
```

Copy the file srs.secret between all cluster/mx servers so it matches on all systems, this file contains a random number which is used to verify srs and allow emails. This file will only exist once you start using SPF, it resides in the g_home directory.

On the 'front end' servers

Don't define the real domain names on the front end server, it will just use gateway settings. Use a sub domain of the real domain, e.g. "incoming.my.domain"

```
g_gateway domain="fred.com" to="backend.ip" check="true"  
g_gateway domain="xyz.com" to="backend.ip" check="true"  
g_spf_skip "ip.of.backend.server*"  
g_spf_share "ip.of.all.other.servers" (this is a list, not a wild card)
```

NEVER implement front end filters like this unless you have to, it is always a mistake to have front end servers unless your load leaves you with no other choice! Unlike most situations it is not wise to start with this layout because you think your load 'might' require it one day!!!

On webmail servers

Don't define the domain at all in surgemail.ini (just use some fake domain name)

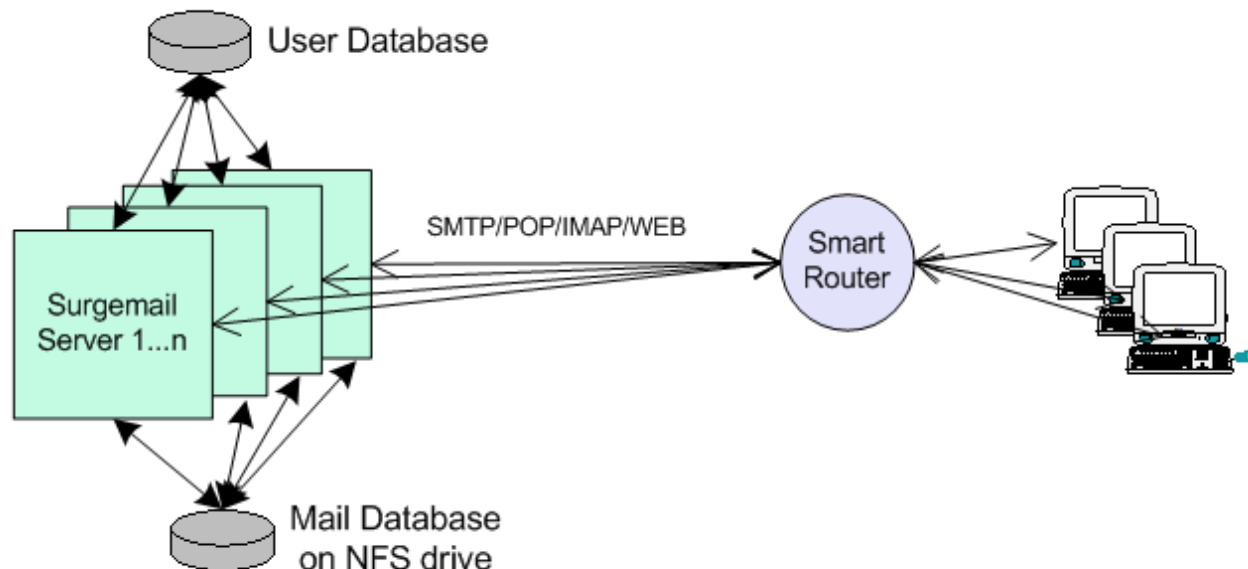
In webmail.ini set:

```
imaphost backend.ip.address  
smtp host backend.ip.address  
use_id_autologin true
```


Configuring 'Shared Storage Clusters'

SurgeMail can be configured in a more traditional shared storage cluster configuration using an NFS (or other) shared storage device for providing standard mail services.

In this configuration you have several servers all running surgeMail handling all mail services storing users mail using the same central storage. The incoming connection load is shared between all servers using an appropriate technique. This is typically a hardware based load balancing router.



STOP! we think this is the wrong solution for you - here's why

The normal reason for using this type of cluster are:

- To get load sharing
- So you can add servers as needed
- To create a fault redundant reliable system

All of these reasons are invalid or better done using surgeMail's mirroring, here's why:

- Load sharing & Adding servers - this assumes the limiting factor is the cpu, due to the speed of modern multi processor systems, this is not the limiting factor, the disk IO is probably going to be the limiting factor, so you gain nothing with the extra boxes. This is because only improving the 'bottle neck' increases throughput.
- Redundant failover etc - since only a single copy of the data exists on the shared network drive, the system is not particularly redundant. Your network drive may be raid or something else fancy, but it's still a single point of failure - and network fancy drives are significantly more prone to failure than simple internal disk arrays, this is just a feature of complexity, always avoid complexity!
- Performance - network drives tend to be significantly slower, this is because small requests that would be cached on a local drive, cannot be cached and must go over the network link. Even if the network link can cope with the level of requests (and the limit is packets per second, not bytes per second so a faster network may not help)
- Complexity/reliability - the shared drive approach adds significantly to your systems complexity, as a result failures are very likely to be more common.
- Some features of SurgeMail will not fully work in this environment. (SurgePlus for example can only be used on a single cluster member) Core functionality should work fine though.
- In summary, although SurgeMail supports this layout, we have almost never seen a system where it was the 'right' solution to use, careful use of mirroring and split functionality will almost certainly give you a more reliable, faster, simpler system for less cost.
- But there are exceptions, so if you are sure, proceed :-)

How to configure surgemail to use a shared NFS (or network) drive:

- You should only share the 'mailbox_path' for each domain, do not share the workarea or other directories within surgemail.
- g_share_quota "true"
- g_share_mail "true"
- g_pop_lock "true"
- If you are using a smart router or load balancing tool, [read this page](#)
- Do not use of the automatic account expiry and deletion functions such as g_delete_user_after / g_delete_user_mode / g_delete_user_suspend functions.
- To allow webmail to be run on more than one host [see this page](#).
- In the surgemail home directory sym link the directory 'resp' to a shared area (this allows the auto responder to work correctly and each cluster member will remember who has been replied to)
- g_autologin_file "/shared/autologin.dat" - Where that file is shared between all servers via nfs
- In addition you probably should share the directory (suregmail home)/resp (We will provide mechanism to do this on windows shortly)
- g_domuser_file "/shared_folder/domuser.dat" - define the alias file in a shared area so user aliases work

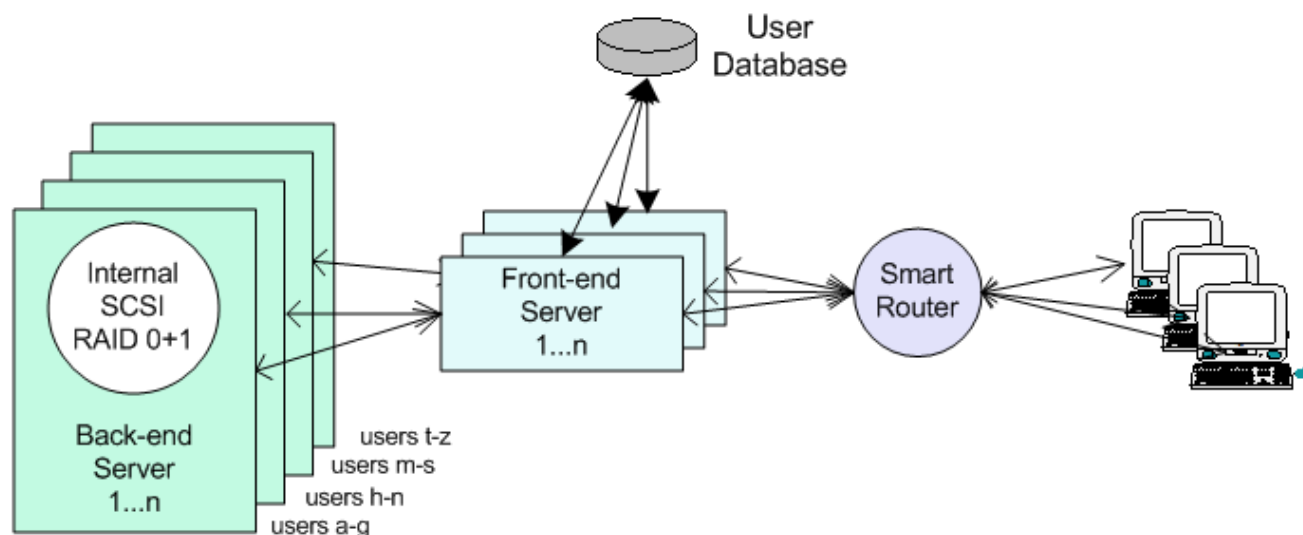
Warnings - Things to avoid doing in this configuration:

- You cannot use surgemail 'mirroring' feature in addition to sharing the mail spool.
- Do not use the migration mechanism on multiple shared hosts at the same time, as then it will be in danger of converting one user twice at the same time.

Configuring 'Proxy Mode Clusters'

This system allows both infinite scaling, and 3 layer security. The incoming POP/SMTP connections arrive at one of several front end 'proxy' servers (running SurgeMail in proxy mode) these servers then lookup the user in the networked user database (via LDAP or our own TCPAuth module) and along with the normal response an extra response code of 'tohost=backend.host.name' is returned, the proxy then redirects the user to the appropriate back end system.

So you might run 4 back end systems, each with 100,000 users, and 2 front end systems. To add more users you just add as many front end and back end servers as needed to cope with the load.



Each user is only on one of the back end systems, the only piece in the system that has to handle all the users is the user database, which is a relatively trivial task as the quantity of data per entry is so small. We recommend the use of NWAAuth or LDAPAuth but any of the database back end authentic modules would be suitable.

[See here for technical details](#)

Note: 3 Layer Security: This model is called '3 layer security' as the front and back end systems can be separated by another fire wall. And in the case of 'WebMail' the user web interface can also be separated from the front end systems by a fire wall, hence '3 layer' :-)

To implement this system set on the proxy system the setting `g_proxy true`, and in the authent module add the 'tohost=xxx' field. For existing user accounts you can define `g_proxy_default host.name` so that user records with no 'tohost' entry are correctly sent to the existing back end system. In this way a non proxy based system can be instantly turned into a proxy based system.

IP Failover

Note: This mechanism is **NOT recommended**. We recommend using a smart router to do this or doing manual IP changes as this should be a very rare event. Using scripts as below is not particularly sensible and generally we would always do such changes manually rather than risk a trivial failover event occurring when the backup server is not known to be in sync / tested and properly configured. (as could easily happen during the months between setting this up and a problem occurring)

Incoming connections are routed to a floating IP number, which is not formally assigned to any particular NIC, dynamically configured by SurgeMail. This 'floating' IP is monitored by two actual servers, one of which is configured as the master, and the other as the slave. If the master server goes down, the slave takes over the floating IP. If the slave detects the presence of the master, it relinquishes control of the floating IP. The master will take over the floating IP if it is not already assigned to a server.

This ensures that at any given time the floating IP number is guaranteed to be reachable from the outside world. System Administrators are free to take down either machine for servicing, and SurgeMail will automatically re-assign the floating IP number as needed.

How it works

There are two physical machines involved each of which is running the actual SurgeMail server as well as a monitor daemon. The servers receive failover configuration instructions via the tellmail commands:

```
tellmail failover add ip-number
tellmail failover remove ip-number
```

These commands are typically sent by the monitor daemon but can also be issued from the command line. These commands in turn tell SurgeMail to execute shell-scripts called failover_add.sh and failover_remove.sh respectively (on Win32 platforms, failover_add.bat and failover_remove.bat). These scripts must be placed in the directory SurgeMail was installed to on both the master and slave machines.

These scripts must be edited by the System Administrator as they require system-specific parameters to be set

Additionally, the monitor daemons must also be configured via a failover.conf file, which contains four lines setting various parameters, like this (for the master):

```
failtime 30
ismaster true
aliasip 10.0.0.100
otherip 10.0.0.2
```

or this (for the slave):

```
failtime 30
ismaster false
aliasip 10.0.0.100
otherip 10.0.0.1
```

These config files will set up a failover sharing the floating IP number 10.0.0.100 between a master server with IP number 10.0.0.1 and a slave server with IP number 10.0.0.2. The floating IP is only transferred after *failtime* seconds, to prevent the system overreacting to brief outages.

The Win32 batch files

On Win32 SurgeMail executes the failover_add.bat and failover_remove.bat batch files. They each contain a single line calling the ipalias.exe command. Additional commands such as echoing to a logfile can also be added. The ipalias.exe command works as follows:

ipalias -s*old_ip* -i*alias_ip*

Tells ipalias.exe to search all network interfaces for the first one bound to *old_ip*. It then attaches an alias ip *alias_ip* to that same interface. ipalias.exe is used this way in failover_add.bat. The *alias_ip* argument is passed into the batchfile by Surgemail.

ipalias -r*alias_ip*

Tells ipalias.exe to search all network interfaces for the first one with *alias_ip* attached to it. It then removes the alias from that interface. ipalias.exe is used this way in failover_remove.bat. The *alias_ip* argument is passed into the batchfile by SurgeMail.

Typically only the failover_add.bat file will need to be edited to provide the correct *old_ip* parameter. The ipalias.exe executable should be packaged with the batchfiles.

The Linux shell scripts

On Linux SurgeMail executes the failover_add.sh and failover_remove.sh batch files. They each contain a single line calling the ifconfig command. Additional commands such as echoing to a logfile can also be added. Details on how ifconfig works can be found on the appropriate manpage.

As ifconfig needs root access you must run surgemail as root for these scripts to work. To do this set the ownership of the home surgemail directory, e.g.

```
chown root /usr/local/surgemail
```

ifconfig eth0:1 alias_ip

Tells ifconfig to bind the given *alias_ip* to the given ethernet adapter and unit number. The *alias_ip* argument is passed into the batchfile by SurgeMail. Both a driver name and a unit number must be supplied.

ifconfig eth0:1 down

Tells ifconfig to unbind the ethernet adapter/unit number. The driver name and unit number *must* be the same as those specified in *failover_add.sh*.

Both shell scripts will need to be edited to provide the correct parameters to ifconfig. A full path may also need to be specified.

Download Scripts + Documentation

Version 1.0	
download	Microsoft Windows (9x,ME,NT,2000,XP) + Linux (libc6)

Sorry this product is no longer available. We recommend you try a web monitoring service instead.

(The rest of this page is only here for existing customers and historical reference)

The essential network services monitor, Internet Watch Dog can monitor a virtually unlimited number of services on your TCP/IP network and notify of alarm conditions — anytime, anywhere. If you're the person responsible for an ISP, a local Intranet or just a user who wants evidence to show how well or badly the system is working, then you need Internet WatchDog monitoring your services 24 hours a day. When your Internet/Intranet services (web, email, POP, news, FTP, telnet) stop responding Internet Watch Dog will let you know first (it will even page you). No more worrying that a service can go down on your system and you won't know about it until your users tell you. Internet Watch Dog's Reports will pin point any problem for you so that you don't waste time trouble shooting, any problem can be fixed efficiently with minimal downtime and inconvenience for your users or loss to your company. Internet Watch Dog is not just a great, low cost, easy to use utility, its one you can't afford to be without.

[Click here to Download Now](#)

Features

- [Watch Dog can test web pages, email services, FTP, SMTP, POP, echo, proxy, news etc...](#)
- [Provides a regular report to show your boss or customers](#)
- **Now** - [Web page reports](#) !
- [Gives TCP/IP logs showing the cause of any problem](#)
- [Calls your pager or phone to let you know what's wrong](#)
- **New** - [Alpha-Numeric](#) pager support (version 3.4, beta)
- [Plays sounds and gives visual display of a problem](#)
- [Tests can be scheduled at any frequency required](#)
- [Fast and efficient support](#)
- [How to purchase?](#)
- [Year 2000 Compliance Certificate](#)
- [Uninstall Instructions](#)
- [Beta Versions and Updates](#)

Watch Dog can test web pages, email services...

Internet Watch Dog allows you to test almost any TCP/IP based service, Watch Dog opens a connection to the specified host and port, and then sends a command, it then waits for a specified response, if the response is not correct, or the host cannot be reached, then the 'action' is initiated for that test.

For each test you can define any of several actions, see [Actions Internet Watch Dog can perform](#)

[Back to Top](#)

Regular reports

Because Watch Dog is monitoring your system 24 hours a day it can provide email reports showing useful statistics on all services. **NOW** with versions 3.3 upwards WatchDog can also produce its reports as a fully customizable [Web Page](#).

You may know that your service runs 99.98% of the time, but do your users know that? does your boss? do they believe you? With Internet Watch Dog you can have regular reports showing how well the system is running, this may be the most valuable feature of Internet Watch Dog.

Alternatively, maybe your company has web pages and an email address on the internet, but how do you know if they are really available and working 24 hours a day. You need to know if the web service that hosts your pages is reliable, it will make your company look very bad if your web pages keep disappearing!!!

Here is an example of an email report from Internet Watch Dog:

Date: Thu, 19 Feb 1970 16:08:40							
Host:Port	Bad	Good	Delay	Best	Worst	Uptime	
-----	---	---	-----	---	---	---	
167.29.2.5:SMTP	23	2440	4.4	0	49	99.07%	

167.29.2.5:POP	2489	0.1	0	17	100.00%	
167.29.2.5:EMAIL	5	43	19.1	3	367	89.58%

Times of last five failures:

- Thu 31-Oct 12:59:54 3
- Thu 31-Oct 14:39:57 3
- Thu 31-Oct 14:41:58 3
- Thu 31-Oct 15:47:21 3
- Tue 05-Nov 15:54:43 3

[Back to Top](#)

Web Page Reports - NEW

WatchDog can also produce a web page version of the report from a fully customizable template (version 3.3 upwards).

You simply provide WatchDog with an HTML template file, similar to the ones used by Netwin's DMailWeb and CWMail , and it will create an HTML version of the report whenever it generates a report email.

The WatchDog help file provides a list of variables and simple commands which you can put in your template file . As WatchDog creates the HTML version of the report it replaces the variables as per the command instructions with the required data.

An example template file is provided for you to use as is, or to quickly modify to get the look that you need to impress your audience!

[Back to Top](#)

Actions Internet Watch Dog can perform

When an error occurs Internet Watch Dog can perform any of the following actions:

- Bring the Watch Dog window to the top
- Play any specified sound
- Send an email message
- Phone a pager or normal phone.
- **New** - Alpha-Numeric pager support with version 3.4 (beta)
- Run any command or batch file

Additionally, when a previously failing test works, it can also call your pager with a different code or send a different email message to let you know the problem is solved.

[Back to Top](#)

Now with version 3.4 (beta) you can also send both failure and fixed notification messages to your Alpha-Numeric pager.

The ability to run commands actually allows Internet Watch Dog to repair some problems, on Windows NT you might restart a service using the command:

```
net start smtp
```

What hardware do I need to make paging possible?

All you need is a standard PC with a standard modem. A mini page beeper that can display a different numeric code for each service will typically cost approx. \$10 a month to operate (this will vary depending on your local telecom service)

For Alpha-Numeric paging, you need a pager capable of receiving text messages rather than one which only displays preset messages. You will also need to know the access telephone number to your paging system for a modem using the TAPI (also known as IXO) standard.

[Back to Top](#)

TCP/IP logs showing the problem

If a problem does occur you can quickly display a detailed log of the TCP/IP transaction, Internet Watch Dog retrieves the relevant section of the log for you. You can also re-test the problem service and watch an updating log screen as the connection is made so you can see the cause of the problem quickly.

Test details from every test are logged in one overall log file. You can control the maximum size of the log with a configuration setting. On reaching this setting WatchDog will rotate the logs for you, always saving the last 4. You then have plenty of time to move the previous log files if they need to be stored for future reference(version 3.1b upwards).

[Back to Top](#)

Tests can be scheduled at any frequency required

Internet Watch Dog lets you schedule each test at different time intervals, for example you can run tests:

- Every minute
- Every 3 minutes
- Every hour
- Every 3 days
- Every month
- At 10:20a.m.
- At 5:00a.m.

[Back to Top](#)

Fast and efficient support

Like all of Netwin's products, Internet Watch Dog comes with fast and efficient email support from our expert team.

Simply email: support-watchdog@netwinsite.com

To get the problem resolved faster you might like to send us the following files with your initial question, if they exist on your machine. They are all reasonably small.

from the WatchDog directory:

wdog.itm
wdog.rec
wdog.log, wdog1.log ... etc.
wdog.snd (not often there)

from the system directory (i.e. \winnt or \windows):

wdog.ini

If you want to point out a specific problem then it helps if you can catch it in the log file. To achieve this you may find that one of the two methods detailed below might suit the problem that you are experiencing.

Do the test or action that results in the problem and then, shutdown Watch Dog straight away so that the log is closed ready to send to us, or copy the log to another file before Watch Dog has had a chance to add something else to it.

If you are not sure what action causes the problem then try setting the log rotate size in the 'Config' dialog to something large like 1 Mbyte and then run Watch Dog for a while (like 5-10mins) doing different tests etc so that there is a good chance of getting a useful log.

[Back to Top](#)

How to purchase?

Your Internet Watch Dog includes free updates and support by email for 3 months.

After you have installed the WatchDog product, run it and then use the 'REGISTER' button on the main dialogue window. This creates a registration file called register.txt from the answers that you provide and places it in Watch Dog's working directory. It also copies itself to the clipboard for easy pasting into an email message.

Then Email the registration file, register.txt to sales@netwinsite.com OR your Internet Watch Dog Reseller.

Note - License life time:

Your Internet Watch Dog license allows you to continue using your version of the software **forever**, it does not expire. However if you wish to update to **new versions** released after the support period then you must purchase a new license.

Payment

Payment is normally made by Credit Card. We can accept payment via Visa, Master Card and American Express. Simply enter the name as it is specified on the card, the card number and expiry date into the Registration Form. This data is encrypted for protection. However if you prefer you can FAX your details to us on +64 6 353 7359.

[Back to Top](#)

Uninstall Instructions

To uninstall WatchDog click on 'Config' then 'Delete Service', then simply delete the one WatchDog directory and the wdog.ini file from your system directory.

On Windows 95 all you need do is remove the shortcut from the startup folder and delete the one WatchDog directory, along with the wdog.ini file from your system directory (e.g. c:\windows).

[*Click here to Download Now*](#)

[Back to Top](#)

Templates

Templates files are the core of the CGI's flexibility. The templates are used to generate the look and feel of the product for your customers. The templates are basically HTML code with our '||variables||' and '||cmds||'.

WebMail comes with a few templates sets when you download and install it. These are: 'Panel', 'Smooth', and 'Surge', with various color and language options for "Smooth" and "Surge".

Each template set was designed with a difference purpose in mind but all are easy to use.

- "Surge" is fast, simple, multi language and has few images.
- "Smooth" is image rich, multi language and great for fast connections
- "Panel" is an earlier, but updated template set with proven reliability

Many customers have changed WebMail's look completely, and a few examples of these sites can be seen in our [Gallery](#).

'Panel' and 'Smooth' tpl sets are complex and we suggest that, unless you are sure about javascript and the problems between different browsers (IE / Netscape / Opera etc.), that you do not change these tpl sets. If you you wish to generate your own template design we suggest that you use the 'Surge' as a base reference.

Template Files

Different WebMail template sets can use different template files names. We have generated a simple guide on some of the default templates that webmail comes with.

- [Panel, Surge and Smooth Template Guides](#)

Below is a list of the default template files, and a brief description of what each is for. Each file has the option of having a frame version, which has the same name with the suffix f for frames. For a frame login the frame.tpl is loaded and menubar.tpl is placed in the top frame.

File Name	Description
addrbook.tpl	For editing/adding address books
attach.tpl	For adding/deleting/sending attachments.
bookonly.tpl	Pop up address book (called from send.tpl)
bulletin.tpl	This shows the list of bulletins which the user is allowed to see
ch_pass.tpl	This is used to change passwords when using POPPASSD. See the link Changing Passwords for more details
config.tpl	User Configuration Page (or options page)
confirm.tpl	The template that is displayed when a confirm message has been sent.
confirm.msg	The body of the confirm message that is sent out.
del_fail.tpl	When the user runs out of disk quota and tries to delete Email, this template is displayed.
email.tpl	The popup Email page. Used on the forwarding page.
error.tpl	Used to display any error messages
external.tpl	This is the page which is used when displaying global external address books
folder.tpl	Folder management
forward.tpl	For forwarding one or more messages
group.tpl	Used for creating a list of Emails.
help.tpl	User Manual
item.tpl	A single mail item
large.tpl	This template is used when a large Email is detected and needs to be downloaded. This template has the same layout as the item.tpl.
list.tpl	A list of new mail messages
login.tpl	The first page shown when WebMail is run. This handles the user login
logout.tpl	Logout Screen to be displayed
mail.tpl	This template is used for automatic login.
manager.tpl	This template is the manager's option page.
menubar.tpl	Menu bar is used if using Frame Login rather than normal login
new_msg.tpl	This is the pop up list of new messages only.
newuser.tpl	This is the first template that New Users see (defaults to list.tpl is not available)

nfmenu.tpl	This is the non-frame menubar, that the non-frame templates include.
ok.tpl	Used to display any successful actions, eg Mail sent
pick.tpl	Displays pick list, distribution list and address book
search.tpl	For searching for Emails.
send.tpl	For sending a new mail message
send_bull.tpl	This is the template that is used to create new bulletins
sp_dic.tpl	This shows the words in your private dictionary
sp_fix.tpl	This template, in conjunction with sp_show.tpl, is used to change spelling mistakes, or add words to the private dictionary. (java script only)
sp_show.tpl	This is used to show spelling errors, if there are any. (java script only)
stud.tpl	This is the popup INBOX that reloads after 5 mins.
todo.tpl	This template is used to create items on the user's todo list.

There are other example sets of template files available. If you create a particularly nice set you might like to share them with others. If so, send them to netwin@netwinsite.com and we will add them to this list.

CGI Commands

The template files contain ordinary html plus `||variables||`. These variables are setup by the CGI commands and template commands. The CGI commands may take the form of a submit button with a particular name, a hidden cmd (or xcmd) field inside a form or as a query in the form `cmd=xyz`

Every CGI cmd can be set up as a button, hidden or as part of a query. When commands are set up as buttons there must be no hidden command field in the form, or this will override the button.

eg. Button:	<code><input type=submit name="xxxxxx" value="any thing"></code>
Hidden Field:	<code><input type=hidden name="cmd" value="xxxxxx"></code>
Query:	<code>any thing</code>

The list of valid CGI commands is show below. This list also shows which default template is displayed. The input and output variables that are needed/setup are also displayed for each command. In addition, some commands change user settings are also displayed.

Any variable in the lists that start with a '~' is an ini setting.

CMD	Function	Template												
url	This will make the CGI display the link provided instead.	(CGI)												
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>url</td><td>The 'http' link that you want the CGI to display.</td></tr></table>			Input Variables:		url	The 'http' link that you want the CGI to display.								
Input Variables:														
url	The 'http' link that you want the CGI to display.													
register	Displays the registering screen.	(CGI)												
manager	Displays the manager's screen.	(manager.tpl)												
login	This will login a user.	(frame.tpl or list.tpl)												
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>user</td><td>The user name to login to.</td></tr><tr><td>pass</td><td>The user's password.</td></tr><tr><td>host</td><td>The host that the user wishes to connect to. (optional)</td></tr><tr><td>tcode</td><td>This is used to determine whether the login page is fresh. You must have this unless you have: no_tcode true in your ini file</td></tr><tr><td colspan="2">Output Variables:</td></tr></table>			Input Variables:		user	The user name to login to.	pass	The user's password.	host	The host that the user wishes to connect to. (optional)	tcode	This is used to determine whether the login page is fresh. You must have this unless you have: no_tcode true in your ini file	Output Variables:	
Input Variables:														
user	The user name to login to.													
pass	The user's password.													
host	The host that the user wishes to connect to. (optional)													
tcode	This is used to determine whether the login page is fresh. You must have this unless you have: no_tcode true in your ini file													
Output Variables:														

_played_sound	Setup to '0' if the sound has not played.
utoken	The user's token, which must be passed to CGI every time.
User settings:	
_selected_tpl	This is used to determine which templates the user is using.
wml_agent	The last wml agent used.
agent	The last web browser used.
frames	Set to '1' if the user is using the frames templates.

quick_login	This will perform a login without connecting to the mail server to get Emails.	(frame.tpl or list.tpl)
-------------	--	-------------------------

Input Variables:	
same as 'login'	
Output Variables:	
same as 'login'	
User settings:	
same as 'login'	

auto_login	This will perform an auto-login for a user.	(frame.tpl or list.tpl)
------------	---	-------------------------

Input Variables:	
same as 'login'	
Output Variables:	
same as 'login'	
User settings:	
same as 'login'	

netwin_login	This command is used to go between NetWin products.	(url)
--------------	---	-------

Input Variables:	
utoken	The user's utoken.
~netwin_autologin	The list of information required to perform a NetWin login.

logout	This will logout a user.	(logout.tpl)
--------	--------------------------	--------------

Input Variables:	
utoken	The user's utoken.

logout_go	This will logout a user, and then go directly to a URL.	(url)
-----------	---	-------

Input Variables:	
utoken	The user's utoken.
url	The url which the user should be sent to.

menubar	This will display the menubar. Used in frame.tpl only	(menubar.tpl)
---------	---	---------------

Input Variables:	
utoken	The users utoken.

reload_mail	This will check your INBOX for new mail.	(list.tpl)
-------------	--	------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(optional: defaults INBOX)</i>
Output Variables:	
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

check_mail	This will display your INBOX.	(list.tpl)
------------	-------------------------------	------------

Input Variables:	
utoken	The user's utoken.
Output Variables:	
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

list	This will list the folder provided.	(list.tpl)
------	-------------------------------------	------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(optional: defaults INBOX)</i>
timezone	The timezone which all the messages are displayed in. <i>(optional)</i>
sort_on	This is the field to sort on. ie. 'Subject:', 'h_bytes', 'Date:' <i>(optional)</i>
sort_reverse	When set to 'true', the order of the sort is reverse.
sort_method	The method used to compare the values. ie. 'text', 'date', 'number', 'read_unread', 'new', 'draft'
Output Variables:	
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

soft_list	(exactly the same as 'list', but does not require a lock)	(list.tpl)
-----------	---	------------

Input Variables: - see 'list'	
Output Variables:- see 'list'	

item*	Displays the message.	(item.tpl or large.tpl)
-------	-----------------------	-------------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(optional: defaults INBOX)</i>
timezone	The timezone which the message is displayed in. <i>(optional)</i>
max_email_size	This ini setting is used to determine whether the Email is large and if it is, defaults to displaying the 'large.tpl' instead of the 'item.tpl'.
force_download	When set to true, will always download Email and display the item.tpl.
Output Variables:	

item_id	The position in the list of the Email messages.
v_raw	The user's sticky settings for displaying the Email message in Raw format.
v_headers	The user's sticky settings for displaying the Email message header.
v_font	The user's sticky settings for displaying the Email message in variable width font.
v_inline	The user's sticky settings for displaying the Email message images/html/text attach files inline.
v_show_alt	The user's sticky settings for displaying the Email message's alternative part as well.
show_email	Setup to display the decoded Email.
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

soft_item*	(exactly the same as 'item*', but does not require a lock)	(item.tpl or large.tpl)
------------	--	-------------------------

Input Variables: - see 'item'

Output Variables:- see 'item'

itempart*	Displays the message subpart.	(N/A)
-----------	-------------------------------	-------

Input Variables:

utoken	The user's utoken.
part	Shows which mime part to display.
subpart	Shows which part of the mime part to display.

itemflag*	Changes an item flag status.	(list.tpl)
-----------	------------------------------	------------

Input Variables:

utoken	The user's utoken.
fld/folder	The folder to reload.
flag	The list of flags to change the status to.

saveaddr*	This saves the from/reply Email address to the pick list.	(item.tpl)
-----------	---	------------

Input Variables:

utoken	The user's utoken.
fld/folder	The folder to reload. (optional: defaults INBOX)

saveaddrbook*	This saves the from/reply Email address to the address book.	(addrbook.tpl)
---------------	--	----------------

Input Variables:

utoken	The user's utoken.
fld/folder	The folder to reload. (optional: defaults INBOX)

Output Variables:

abk_email	This is the Email address which the CGI sets up for the address book.
-----------	---

delitem*	This will delete the message.	(list.tpl)
----------	-------------------------------	------------

Input Variables:

utoken	The user's utoken.
--------	--------------------

fld/folder	The folder to reload.
ignore_trash	When 'true', only flags the message for deletion, does not move it to the Trash folder. (optional).
force_empty	When 'true', deletes all messages flagged for deletion in the current folder. These messages are permanently deleted, and do not go to the Trash folder (optional). Often used in conjunction with ignore_trash to delete a message without it going to the trash (eg. ...&force_empty=true&ignore_trash=true&...)

delitem_next*	This will delete the message, and then display the next message.	(<i>item.tpl</i>)
---------------	--	---------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
ignore_trash	When 'true', only flags the message for deletion, does not move it to the Trash folder. (optional).
force_empty	When 'true', deletes all messages flagged for deletion in the current folder. these messages are permanently deleted, and do not go to the Trash folder (optional). Often used in conjunction with ignore_trash to delete a message without it going to the trash (eg. ...&force_empty=true&ignore_trash=true&...)

delsel	This will delete the messages which have been selected.	(<i>list.tpl</i>)
--------	---	---------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
sel_*	The list of items to delete.
ignore_trash	When 'true', only flags the message for deletion, does not move it to the Trash folder. (optional).
force_empty	When 'true', deletes all messages flagged for deletion in the current folder. These messages are permanently deleted, and do not go to the Trash folder (optional). Often used in conjunction with ignore_trash to delete a message without it going to the trash (eg. ...&force_empty=true&ignore_trash=true&...)

moveitem*	This will move the message to the selected folder.	(<i>list.tpl</i>)
-----------	--	---------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
dstfld	The destination folder name, where the Email message will be moved.

movesel	This moves the selected messages to a folder.	(<i>list.tpl</i>)
---------	---	---------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
sel_*	The list of items to delete.
dstfld	The destination folder name, where the Email message will be moved to.

copyitem*	This copies the message to the selected folder.	(<i>list.tpl</i>)
-----------	---	---------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(optional: defaults INBOX)</i>
dstfld	The destination folder name, where the Email message will be copied to.

copysel	This copies the selected messages into a folder.	<i>(list.tpl)</i>
---------	--	-------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(optional: defaults INBOX)</i>
sel_*	The list of items to copy.
dstfld	The destination folder name, where the Email message will be copied.

forward_edit*	This forwards the message, allowing editing of the Email message. It also preserves the attachments.	<i>(send.tpl)</i>
---------------	--	-------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
Output Variables:	
This 'forward_edit' command has the same output variables as the 'msg_reply' command.	

forward*	This forwards the message.	<i>(forward.tpl)</i>
----------	----------------------------	----------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
Output Variables:	
reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.
fwd_list	The list of Email UIDL's that are to be forwarded.
n fwd	The number of Emails which are being forwarded.
v_myfrom	The default which decides whether or not the user's from address will be used.
v_supheader	This setting is used to determine whether the headers of the Email should be suppressed.
picklist	The list of Email addresses in the pick list and the distribution list.

forwardsel	This forwards the selected messages.	<i>(forward.tpl)</i>
------------	--------------------------------------	----------------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
sel_*	The list of items to delete.
Output Variables:	
fwd_list	The list of Email UIDL's that are to be forwarded.

nfwd	The number of Emails which are being forwarded.
v_myfrom	The default which decides whether or not the user's from address will be used.
v_supheader	This setting is used to determine whether the headers of the Email should be suppressed.
picklist	The list of Email addresses in the pick list and the distribution list.

forward_send	This sends the forwarded message.	(ok.tpl)
--------------	-----------------------------------	----------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
fwd_list	The list of Email UIDL's that are to be forwarded.
fwd_note	The note that will be attached to the Email being forwarded.

forward_senddel	This sends the forwarded message and deletes the message.	(ok.tpl)
-----------------	---	----------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload.
fwd_list	The list of Email UIDL's that are to be forwarded.
fwd_note	The note which will be attached to the Email which is being forwarded.

msg_save*	This will save the Email to disk.	(N/A)
-----------	-----------------------------------	-------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. (optional: defaults INBOX)

msg_reply*	This replies to the message's 'From' field.	(send.tpl)
------------	---	------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. (Optional: defaults INBOX)
reply_prefix	The prefix of the message. (Optional: defaults '> ')
Output Variables:	
to cc	The unchanged 'To:' and 'Cc:' header of the Email
from from_email from_personal	Shows who the Email came from. This is where you can display whether the address uses the personal, the complete or the Email format. ie. Complete: "Lynden" <lynden@1.2.3.4> Email: lynden@1.2.3.4 Personal: "Lynden"
subject	The subject of the Email.
date	The date of the Email
h_from h_from_email	The From: field where the message should go.

h_from_personal	
h_subject	The default subject of the reply Email.
quote_body	The body of the Email, converted to display inside the html page.
reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.
host_user* host_name*	Used for the pull down list of multiple host accounts.
addsig	Set to 'checked' if the user sent a signature the previous time.
send_autocc	Set to 'checked' if the user sent a copy to herself the previous time.
copyself	Set to 'checked' if the user saved in the sent folder the previous time.
pick_item	The list of items in the pick list and distribution list.
picklist	The list of items in the pick list and distribution list.

msg_replyall*	This replies to a message's 'To, CC, BCC' fields.	(send.tpl)
---------------	---	------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(Optional: defaults INBOX)</i>
reply_prefix	The prefix of the message. <i>(Optional: defaults '> ')</i>
Output Variables:	
to cc	The unchanged 'To:' and 'CC:' header of the Email
from from_email from_personal	Shows who the Email came from and who to reply to. This is where you can display whether the address uses the personal, the complete or the Email format. ie. Complete: "Lynden" <lynden@1.2.3.4> Email: lynden@1.2.3.4 Personal: "Lynden"
subject	The subject of the Email.
date	The date of the Email
h_cc	The list of Emails that should be CC'ed to.
h_from	The From: field where the message should be sent.
h_subject	The default subject of the reply Email.
quote_body	The body of the Email, converted to display inside the html page.
reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.
host_user* host_name*	Used for the pull down list of multiple host accounts.
addsig	Set to 'checked' if the user sent a signature the previous time.
send_autocc	Set to 'checked' if the user sent a copy to herself the previous time.
copyself	Set to 'checked' if the user saved in the sent folder the previous time.
pick_item	The list of items in the pick list and distribution list.
picklist	The list of items in the pick list and distribution list.

msg_draft*	This treats the message as a draft and sets up the Email to be edited and sent.	(item.tpl)
------------	---	------------

Input Variables:	
utoken	The user's utoken.
fld/folder	The folder to reload. <i>(Optional: defaults INBOX)</i>
Output Variables:	
h_from	The From: field where the message should be sent.
h_cc	The list of Emails that should be CC'ed to.
h_bcc	The list of Emails that should be BCC'ed to.
h_subject	The default subject of the Email.
quote_body	The body of the Email, converted to display inside the html page.
reply reply_email reply_personal	The default reply of the Email, which is build from the dflt_reply in the user.dat file.
host_user* host_name*	Used for the pull down list of multiple host accounts.
addsig	Set to 'checked' if the user sent a signature the previous time.
send_autocc	Set to 'checked' if the user sent a copy to herself the previous time.
copyself	Set to 'checked' if the user saved in the sent folder the previous time.
pick_item	The list of items in the pick list and distribution list.
picklist	The list of items in the pick list and distribution list.

msg_new	This displays a new message to send.	(send.tpl)
---------	--------------------------------------	------------

Input Variables:	
utoken	The user's utoken.
keep_attach	When set to 'true', the save attachments are not cleared. <i>(Optional: defaults false)</i>
Output Variables:	
reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.
host_user* host_name*	Used for the pull down list of multiple host accounts.
addsig	Set to 'checked' if the user sent a signature the previous time.
send_autocc	Set to 'checked' if the user sent a copy to herself the previous time.
copyself	Set to 'checked' if the user saved in the sent folder the previous time.
pick_item	The list of items in the pick list and distribution list.
picklist	The list of items in the pick list and distribution list.

msg_search	Displays the Email search page.	(search.tpl)
------------	---------------------------------	--------------

<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr></table>			Input Variables:		utoken	The user's utoken.																		
Input Variables:																								
utoken	The user's utoken.																							
dosearch	Performs the Email search.	(search.tpl)																						
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>search_body</td><td>The search string for the body of the Email. (Optional)</td></tr><tr><td>search_body_size</td><td>The number of characters in the body of the Email to search. (Optional)</td></tr><tr><td>search_subject</td><td>The search string in the subject field.</td></tr><tr><td>search_from</td><td>The search string in the from field.</td></tr><tr><td>search_mode</td><td>When set to 'true', the search is an 'OR' search.</td></tr><tr><td>multi_sel_fld</td><td>The list of folders which the search is to be performed in.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td colspan="2">The results from the search are generated by the begin_search ... end_search command.</td></tr></table>			Input Variables:		utoken	The user's utoken.	search_body	The search string for the body of the Email. (Optional)	search_body_size	The number of characters in the body of the Email to search. (Optional)	search_subject	The search string in the subject field.	search_from	The search string in the from field.	search_mode	When set to 'true', the search is an 'OR' search.	multi_sel_fld	The list of folders which the search is to be performed in.	Output Variables:		The results from the search are generated by the begin_search ... end_search command.			
Input Variables:																								
utoken	The user's utoken.																							
search_body	The search string for the body of the Email. (Optional)																							
search_body_size	The number of characters in the body of the Email to search. (Optional)																							
search_subject	The search string in the subject field.																							
search_from	The search string in the from field.																							
search_mode	When set to 'true', the search is an 'OR' search.																							
multi_sel_fld	The list of folders which the search is to be performed in.																							
Output Variables:																								
The results from the search are generated by the begin_search ... end_search command.																								
add_attach	Displays the attached Emails.	(attach.tpl)																						
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>attach_num</td><td>The number of attachments which are going to be sent.</td></tr></table>			Input Variables:		utoken	The user's utoken.	Output Variables:		attach_num	The number of attachments which are going to be sent.														
Input Variables:																								
utoken	The user's utoken.																							
Output Variables:																								
attach_num	The number of attachments which are going to be sent.																							
attach_msg	This will add a message to the list of attachments.	(attach.tpl)																						
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>attach</td><td>The list of attached files.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>attach_num</td><td>The number of attachments which are going to be sent.</td></tr></table>			Input Variables:		utoken	The user's utoken.	attach	The list of attached files.	Output Variables:		attach_num	The number of attachments which are going to be sent.												
Input Variables:																								
utoken	The user's utoken.																							
attach	The list of attached files.																							
Output Variables:																								
attach_num	The number of attachments which are going to be sent.																							
attach_send	This will display back the Email message which is to be sent.	(send.tpl)																						
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>h_cc</td><td>The list of Emails that should be CC'ed to.</td></tr><tr><td>h_from</td><td>The From: field where the message should be sent.</td></tr><tr><td>h_subject</td><td>The default subject of the reply Email.</td></tr><tr><td>Quote_body</td><td>The body of the Email, converted to display inside the html page.</td></tr><tr><td>reply reply_email reply_personal</td><td>The default reply of the Email, which is built from the dflt_reply in the user.dat file.</td></tr><tr><td>host_user* host_name*</td><td>Used for the pull down list of multiple host accounts.</td></tr><tr><td>addsig</td><td>Set to 'checked' if the user sent a signature the previous time.</td></tr><tr><td>send_autocc</td><td>Set to 'checked' if the user sent a copy to self the previous time.</td></tr></table>			Input Variables:		utoken	The user's utoken.	Output Variables:		h_cc	The list of Emails that should be CC'ed to.	h_from	The From: field where the message should be sent.	h_subject	The default subject of the reply Email.	Quote_body	The body of the Email, converted to display inside the html page.	reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.	host_user* host_name*	Used for the pull down list of multiple host accounts.	addsig	Set to 'checked' if the user sent a signature the previous time.	send_autocc	Set to 'checked' if the user sent a copy to self the previous time.
Input Variables:																								
utoken	The user's utoken.																							
Output Variables:																								
h_cc	The list of Emails that should be CC'ed to.																							
h_from	The From: field where the message should be sent.																							
h_subject	The default subject of the reply Email.																							
Quote_body	The body of the Email, converted to display inside the html page.																							
reply reply_email reply_personal	The default reply of the Email, which is built from the dflt_reply in the user.dat file.																							
host_user* host_name*	Used for the pull down list of multiple host accounts.																							
addsig	Set to 'checked' if the user sent a signature the previous time.																							
send_autocc	Set to 'checked' if the user sent a copy to self the previous time.																							

copyself	Set to 'checked' if the user saved in the sent folder the previous time.
pick_item	The list of items in the pick list and distribution list.
picklist	The list of items in the pick list and distribution list.

attach_del	This will delete the highlighted message	(attach.tpl)
------------	--	--------------

Input Variables:	
utoken	The user's utoken.
selected_attach	The selected list of attached files to remove.
Output Variables:	
attach_num	The number of attachments which are going to be sent.

send	This sends the message	(ok.tpl)
------	------------------------	----------

Input Variables:	
utoken	The user's utoken.
to	The list of people that the message should be sent to.
CC	The list of people that a carbon copy should be sent to. (Optional)
BCC	The list of people that a blind carbon copy should be sent to. (Optional)
attach	The list of attached files. (Optional)
reply	The reply-to field of the Email message. (Optional)
from	The From: header of the Email. (Optional, defaults to user@domain)
subject	The subject of the Email. (Optional)
message	The body of the Email message. (Optional)
addsig	If 'checked', a signature will be sent. (Optional)
send_autocc	If 'checked, a copy will be sent to self. (Optional)
copyself	If 'checked', a copy will be saved in the sent folder. (Optional)
priority x-priority	This is the text priority level of the Email. (Optional)
content- type	This is the content-type of this message. (Optional)
confirm	When set to 'read', the message will be sent flagged, so that the recipient will be reminded to reply, telling you that the message has been read. (Optional)

save_draft	Save the Email message as a draft message.	(ok.tpl)
------------	--	----------

Input Variables:	
utoken	The user's utoken.
dstfld	The folder which the draft message will be placed in.
to	The list of people that the message should be sent to.
CC	The list of people that a carbon copy should be sent to. (Optional)
BCC	The list of people that a blind carbon copy should be sent to. (Optional)
attach	The list of attached files. (Optional)
reply	The reply-to field of the Email message. (Optional)

subject	The subject of the Email message. <i>(Optional)</i>
message	The body of the Email message. <i>(Optional)</i>
addsig	If 'checked', a signature will be sent. <i>(Optional)</i>
send_autocc	If 'checked', a copy will be sent to self. <i>(Optional)</i>
copyself	If 'checked', a copy will be saved in the sent folder. <i>(Optional)</i>
priority	This is the text priority level of the Email. <i>(Optional)</i>
content-type	This is the content-type of this message. <i>(Optional)</i>

no_send	Saves the variables to the user.dat file but does not perform the send.	<i>(send.tpl)</i>
---------	---	-------------------

Input Variables:	
utoken	The user's utoken.
addsig	If 'checked', a signature will be sent. <i>(Optional)</i>
send_autocc	If 'checked', a copy will be sent to self. <i>(Optional)</i>
copyself	If 'checked', a copy will be saved in the sent folder. <i>(Optional)</i>

setview*	This changes how the messages are viewed in item.tpl.	<i>(item.tpl)</i>
----------	---	-------------------

Input Variables:	
utoken	The user's utoken.
v_raw	When set, this shows the message as on the mail server. This overrides the other v_* settings. <i>(Optional)</i>
v_headers	This displays the headers of the Email message. <i>(Optional)</i>
v_font	This displays the message using variable width font. <i>(Optional)</i>
v_inline	This will display html, text and images inline of the Email message. <i>(Optional)</i>
v_exec	This will stop the javascript, applets, forms and scripts from being removed. <i>(Optional)</i>
v_show_alt	This will show the alternative part of the Email message as well. <i>(Optional)</i>
Output Variables:	
Save as the 'item' CGI command.	

config	This displays the user's configuration screen (options page)	<i>(config.tpl)</i>
--------	--	---------------------

Input Variables:	
utoken	The user's utoken.

saveconfig (all) saveconfig_details saveconfig_filter saveconfig_reject_list	This saves the user configuration.	<i>(config.tpl or frame.tpl)</i>
---	------------------------------------	----------------------------------

Input Variables:	
utoken	The user's utoken.

_real_name	The name of the user. <i>(Optional)</i>
_max_message_display	The number of items to display on the list page. <i>(Optional)</i>
new_timezone	The timezone which the user desires to see the messages in. <i>(Optional)</i>
_selected_tpl	The template which is currently being used. <i>(Optional)</i>
_sound_file	The name of the sound file. <i>(Optional)</i>
host_name* host_user* host_pass* host_prefix* host_proifile*	This list of multiple hosts setup. <i>(Optional)</i> (Must have at least host_name0, host_user0) (saveconfig_details - save these)
r_header* r_contains* r_action* r_dstfld* r_addr*	These are the filtering rules. <i>(Optional)</i> (saveconfig_filter - save these)
reject	This is the list of Emails to reject. <i>(Optional)</i> (saveconfig_reject_list - save these)

change_pass	Displays the change password template. (POPPASSD needed) Outdated by NetAuth .	(ch_pass.tpl)
set_pass	Changes the user's password. (POPPASSD needed) Outdated by NetAuth .	(ok.tpl)
fet_add	Adds the fetch account.	(config.tpl)

Input Variables:	
utoken	The user's utoken.
fet_host fet_user fet_pass fet_number	The details of the fetch account. Fetch accounts are accounts that are seen as part of the INBOX.

fet_delete	Delete the selected fetch account.	(config.tpl)
------------	------------------------------------	--------------

Input Variables:	
utoken	The user's utoken.
fet_host fet_user	The details of the fetch account to be removed.

fld_manage	This displays the folder management page.	(folders.tpl)
------------	---	---------------

Input Variables:	
utoken	The user's utoken.
Output Variables:	
fld_inbox_total	The total size of INBOX.
fld_kinbox_total	The total size of INBOX in kbytes
fld_isize_total	The total size of all folders less INBOX.
fld_kisize_total	The total size of all folders less INBOX in Kbytes

fld_create	This creates a new folder.	(folders.tpl)
------------	----------------------------	---------------

Input Variables:	
utoken	The user's utoken.

fld_name	The name of the folder to be created.
Output Variables:	
fld_inbox_total	The total size of INBOX.
fld_kinbox_total	The total size of INBOX in Kbytes
fld_isize_total	The total size of all folders less INBOX.
fld_kisize_total	The total size of all folders less INBOX in Kbytes

fld_delete	This deletes the selected folder.	(folders.tpl)
------------	-----------------------------------	---------------

Input Variables:	
utoken	The user's utoken.
dstfld	The name of the folder to be deleted.
Output Variables:	
fld_inbox_total	The total size of INBOX.
fld_kinbox_total	The total size of INBOX in Kbytes
fld_isize_total	The total size of all folders less INBOX.
fld_kisize_total	The total size of all folders less INBOX in Kbytes.

fld_rename	This renames a selected folder. (dstfld, fld_name)	(folders.tpl)
------------	--	---------------

Input Variables:	
utoken	The user's utoken.
dstfld	The name of the folder to be renamed.
fld_name	The new name of the folder.
Output Variables:	
fld_inbox_total	The total size of INBOX.
fld_kinbox_total	The total size of INBOX in Kbytes
fld_isize_total	The total size of all folders less INBOX.
fld_kisize_total	The total size of all folders less INBOX in Kbytes

pick_edit	This displays the pick.tpl, and has the pick list(recent addresses), distribution list and address book displayed.	(pick.tpl)
-----------	--	------------

Input Variables:	
utoken	The user's utoken.
Output Variables:	
autoadd	This is the setting which determines whether additions to the pick list are done automatically.
picklist	The list of recently seen/used addresses.
dist_list	The distribution list.
abook_edit	The currently selected address book.
gaddr_books	The list of global address books which the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.

pick_save	This will save the changes to the pick list.	(pick.tpl)
-----------	--	------------

Input Variables:	
utoken	The user's utoken.
autoadd	This is the setting which determines whether additions to the pick list are done automatically.
picklist	The list of recently seen/used addresses.
dist_list	The distribution list.

addr_only	This will display the 'bookonly.tpl'	(bookonly.tpl)
-----------	--------------------------------------	----------------

Input Variables:	
utoken	The user's utoken.
abook_edit	The name of the current selected address book.
nick_name	The nickname of the selected record. <i>(Optional)</i>
addr_pos	The position of the first address to start displaying. <i>(optional)</i>
letter_pos	The starting letter of the first address to start displaying. <i>(optional)</i>
~max_wml_display ~_max_message_display	The number of messages to be displayed per page.
Output Variables:	
gaddr_books	The list of global address books that the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.
nick_name abk_*	The nickname of the selected record. This also sends out the rest of the address information for this nick_name.
book_prev	The start number of the previous page.
book_next	The start number of the next page.
name	This variable will create a link to the page used to edit the nickname of the selected record.
script_nick	This variable will create a javascript link to add the nickname to the selected to/cc/BCC field on the send page. (requires the javascript function "pressnick(value)" from the default templates.

create_addr	This will open an Address Book or create an address book.	(addrbook.tpl)
-------------	---	----------------

Input Variables:	
utoken	The user's utoken.
abook_edit	The name of the current selected address book.
nick_name	The nickname of the selected record. <i>(Optional)</i>
Output Variables:	
gaddr_books	The list of global address books which the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.
nick_name abk_*	The nickname of the selected record. This also sends out the rest of the address information for this nick_name.
Name	This variable will create a link to the page used to edit the nickname of the selected record.

compress_addr	This compresses an address book. This is done automatically by the CGI but is available for the user to compress if necessary.	(addrbook.tpl)
---------------	--	----------------

Input Variables:	
utoken	The user's utoken.
abook_edit	The name of the current selected address book.
Output Variables:	
gaddr_books	The list of global address books that the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.
nick_name abk_*	The nickname of the selected record. This also sends out the rest of the address information for this nick_name.
Name	This variable will create a link to the page used to edit the nickname of the selected record.

add_addr	This will add an address book entry to the address book.	(addrbook.tpl)
----------	--	----------------

Input Variables:	
utoken	The user's utoken.
abook_edit	The name of the current selected address book.
nick_name abk_*	The nickname of the selected record. This also sends out the rest of the address information for this nick_name.
Output Variables:	
gaddr_books	The list of global address books which the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.
nick_name abk_*	The nick_name of the selected record. This also sends out the rest of the address information for this nick_name.
Name	This variable will create a link to the page used to edit the nickname of the selected record.

del_addr	This will remove an entry from the address book.	(addrbook.tpl)
----------	--	----------------

Input Variables:	
utoken	The user's utoken.
abook_edit	The name of the currently selected address book.
nick_name	The nickname of the selected record.
Output Variables:	
gaddr_books	The list of global address books that the user makes use of.
addr_books	The list of available address books.
abook	The list of available address books.
nick_name abk_*	The nickname of the selected record. Also sends out the rest of the address information for this nick_name.
Name	This variable will create a link to the page used to edit the nickname of the selected record.

del_abook	This will delete an entire address book.	(addrbook.tpl)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>abook_edit</td><td>The name of the current selected address book.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>gaddr_books</td><td>The list of global address books which the user makes use of.</td></tr><tr><td>addr_books</td><td>The list of available address books.</td></tr><tr><td>abook</td><td>The list of available address books.</td></tr></table>			Input Variables:		utoken	The user's utoken.	abook_edit	The name of the current selected address book.	Output Variables:		gaddr_books	The list of global address books which the user makes use of.	addr_books	The list of available address books.	abook	The list of available address books.
Input Variables:																
utoken	The user's utoken.															
abook_edit	The name of the current selected address book.															
Output Variables:																
gaddr_books	The list of global address books which the user makes use of.															
addr_books	The list of available address books.															
abook	The list of available address books.															
edit_abook	This displays the page where an address book can be edited.	(addrbook.tpl)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>abook_edit</td><td>The name of the currently selected address book.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>gaddr_books</td><td>The list of global address books that the user makes use of.</td></tr><tr><td>addr_books</td><td>The list of available address books.</td></tr><tr><td>abook</td><td>The list of available address books.</td></tr></table>			Input Variables:		utoken	The user's utoken.	abook_edit	The name of the currently selected address book.	Output Variables:		gaddr_books	The list of global address books that the user makes use of.	addr_books	The list of available address books.	abook	The list of available address books.
Input Variables:																
utoken	The user's utoken.															
abook_edit	The name of the currently selected address book.															
Output Variables:																
gaddr_books	The list of global address books that the user makes use of.															
addr_books	The list of available address books.															
abook	The list of available address books.															
empty	This will empty the trash from a trash can or selected folder.	(list.tpl)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr></table>			Input Variables:		utoken	The user's utoken.										
Input Variables:																
utoken	The user's utoken.															
switch	This will change the folder to display in the list.tpl.	(list.tpl)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>dstfld</td><td>The folder to display. (Optional: defaults INBOX)</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td colspan="2">This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.</td></tr></table>			Input Variables:		utoken	The user's utoken.	dstfld	The folder to display. (Optional: defaults INBOX)	Output Variables:		This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.					
Input Variables:																
utoken	The user's utoken.															
dstfld	The folder to display. (Optional: defaults INBOX)															
Output Variables:																
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.																
help	This will display the help.tpl.	(help.tpl)														
show	This is a NULL command which can be used in conjunction with the 'page' command.	(uses page setting)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td>page</td><td>The name of the template which is to be displayed. (Optional)</td></tr></table>			Input Variables:		utoken	The user's utoken.	page	The name of the template which is to be displayed. (Optional)								
Input Variables:																
utoken	The user's utoken.															
page	The name of the template which is to be displayed. (Optional)															
edit_dict	This will display the list of words in your private dictionary.	(sp_dic.tpl)														
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>The user's utoken.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>dict_edit</td><td>The list of words in the user's personal dictionary.</td></tr></table>			Input Variables:		utoken	The user's utoken.	Output Variables:		dict_edit	The list of words in the user's personal dictionary.						
Input Variables:																
utoken	The user's utoken.															
Output Variables:																
dict_edit	The list of words in the user's personal dictionary.															
save_dict	This will save the private dictionary.	(sp_show.tpl)														

Input Variables:		
utoken	The user's utoken.	
dict_edit	The list of words in the user's personal dictionary.	

add_dict	This will add a word to the private dictionary.	(sp_show.tpl)
----------	---	---------------

Input Variables:		
utoken	The user's utoken.	
misspelt_word	a word which is spelt wrong.	
Output Variables:		
The same as the CGI command spell_check		

check_word	This will check one word for spelling.	(sp_fix.tpl)
------------	--	--------------

Input Variables:		
utoken	The user's utoken.	
word	The word that is spelt wrong.	
line	The line number in which the word is spelt wrong.	
nword	The word number in the line in which the word is spelt wrong.	
Output Variables:		
spell_alt	The list of possible correct spellings.	

spell_check	This will check a message for spelling.	(sp_show.tpl)
-------------	---	---------------

Input Variables:		
utoken	The user's utoken.	
message	The message which is to be spellchecked.	
to	The list of people that the message should be sent to. (Optional)	
cc	The list of people that a carbon copy should be sent to. (Optional)	
BCC	The list of people that a blind carbon copy should be sent to. (Optional)	
attach	The list of attached files. (Optional)	
reply	The reply-to field of the Email message. (Optional)	
subject	The subject of the Email message. (Optional)	
addsig	If 'checked', a signature will be sent. (Optional)	
send_autocc	If 'checked', a copy to self will be sent. (Optional)	
copyself	If 'checked', a copy will be saved in the sent folder. (Optional)	
priority	This is the text priority level of the Email. (Optional)	
content-type	This is the content-type of this message. (Optional)	
Output Variables:		
lines_wrong	The number of lines in which spelling mistakes occur.	
nlines	The total number of lines in the Email.	
nwords	The number of words that were checked.	
nwrong	The number of words that are spelt wrong.	
message	The message which is to be spell checked.	
to	The list of people that the message should be sent to.	
CC	The list of people that a carbon copy should be sent to.	

BCC	The list of people that a blind carbon copy should be sent to.
Attach	The list of attached files.
reply	The reply-to field of the Email message.
subject	The subject of the Email message.
addsig	If 'checked', a signature will be sent.
send_autocc	If 'checked', a copy will be sent to self.
copyself	If 'checked', a copy will be saved in the sent folder.
priority	This is the text priority level of the Email.
content-type	This is the content-type of this message.

change_word	This will change a word.	(sp_show.tpl)
-------------	--------------------------	---------------

Input Variables:	
utoken	The user's utoken.
misspelt_word	This is the word which is spelt wrong.
misspelt_line	This is the line number in which the word is spelt wrong.
word_number	This is the place in the line where the word is spelt wrong.
corrected_word	This is the newly corrected word
Output Variables:	
These are the same as the CGI command spell_check	

rebuild_indexes	This re-creates the indexes of the messages.	(list.tpl)
-----------------	--	------------

Input Variables:	
utoken	The user's utoken.

bulletin	This will display the list of Bulletins that are available to view.	(bulletin.tpl)
----------	---	----------------

Input Variables:	
utoken	The user's utoken.
Output Variables:	
folder	The value is 'bulletin_fld', used for viewing bulletin messages

new_bulletin	This is used to create a new bulletin. The user must be allowed to create new bulletins in order for this to work.	(send_bull.tpl)
--------------	--	-----------------

Input Variables:	
utoken	The user's utoken.

send_bulletin	Sends the bulletin.	(ok.tpl)
---------------	---------------------	----------

Input Variables:	
utoken	The user's utoken.
valid_users	The list of users who will get a particular bulletin. (Optional: defaults to all)
valid_day valid_month valid_year	The date on which the bulletin will timeout, and stop being displayed.

to	The list of people that the message will be sent to.
CC	The list of people that a carbon copy will be sent to. <i>(Optional)</i>
BCC	The list of people that a blind carbon copy will be sent to. <i>(Optional)</i>
attach	The list of attached files. <i>(Optional)</i>
reply	The reply-to field of the Email message. <i>(Optional)</i>
subject	The subject of the Email message. <i>(Optional)</i>
message	The body of the Email message. <i>(Optional)</i>
addsig	If 'checked', a signature will be sent. <i>(Optional)</i>
send_autocc	If 'checked', a copy to self will be sent. <i>(Optional)</i>
copyself	If 'checked', a copy will be saved in the sent folder. <i>(Optional)</i>
priority	This is the text priority level of the Email. <i>(Optional)</i>
content-type	This is the content-type of this message. <i>(Optional)</i>

todo	This displays the 'to do' list.	<i>(todo.tpl)</i>
------	---------------------------------	-------------------

Input Variables:	
utoken	The user's utoken.
Output Variables:	
todo_day todo_month todo_year	The day/month/year of today's date.
today_date	Today's date.

todo_add	This adds a 'to do' item to the list.	<i>(todo.tpl)</i>
----------	---------------------------------------	-------------------

Input Variables:	
utoken	The user's utoken.
todo_day todo_month todo_year	This is the word which is spelt wrong.
line	This is the line number in which the word is spelt wrong.
nword	This is the word number in the line in which the word is spelt wrong.
Output Variables:	
spell_alt	This is the list of possible correct spellings.

todo_delete*	Deletes a selected todo.	<i>(todo.tpl)</i>
--------------	--------------------------	-------------------

Input Variables:	
utoken	This is the user's utoken.

profile	Displays the profiles list.	<i>(config.tpl)</i>
---------	-----------------------------	---------------------

Input Variables:	
utoken	This is the user's utoken.
Output Variables:	
_default_profile	This is the current selected profile.
profile_name	This is the current viewed profile.
pro_*	This is the list of variables which are stored in the profile. The default template set has 'pro_sig' which is the user's signature.

profile_add	Adds the profile to the list.	(config.tpl)								
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>This is the user's utoken.</td></tr><tr><td>profile_name</td><td>This is the current viewed profile.</td></tr><tr><td>pro_*</td><td>This is the list of variables which are stored in the profile. The default template set has 'pro_sig' which is the user's signature.</td></tr></table>			Input Variables:		utoken	This is the user's utoken.	profile_name	This is the current viewed profile.	pro_*	This is the list of variables which are stored in the profile. The default template set has 'pro_sig' which is the user's signature.
Input Variables:										
utoken	This is the user's utoken.									
profile_name	This is the current viewed profile.									
pro_*	This is the list of variables which are stored in the profile. The default template set has 'pro_sig' which is the user's signature.									
profile_delete*	Deletes the selected profile.	(config.tpl)								
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>This is the user's utoken.</td></tr><tr><td colspan="2">Output Variables:</td></tr><tr><td>book_allow_edit</td><td>This is setup if the selected book is allowed to be changed.</td></tr></table>			Input Variables:		utoken	This is the user's utoken.	Output Variables:		book_allow_edit	This is setup if the selected book is allowed to be changed.
Input Variables:										
utoken	This is the user's utoken.									
Output Variables:										
book_allow_edit	This is setup if the selected book is allowed to be changed.									
ext_addr	Displays the external address book.	(external.tpl)								
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>This is the user's utoken.</td></tr></table>			Input Variables:		utoken	This is the user's utoken.				
Input Variables:										
utoken	This is the user's utoken.									
ext_search	Searches the external address book.	(external.tpl)								
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>book_selected</td><td>This is the external address book which has been selected.</td></tr><tr><td>ext_*</td><td>These are the search strings for each search field allowed.</td></tr><tr><td>ext_match</td><td>When set to 'true', all search fields have to be matched in order to get a search match.</td></tr></table>			Input Variables:		book_selected	This is the external address book which has been selected.	ext_*	These are the search strings for each search field allowed.	ext_match	When set to 'true', all search fields have to be matched in order to get a search match.
Input Variables:										
book_selected	This is the external address book which has been selected.									
ext_*	These are the search strings for each search field allowed.									
ext_match	When set to 'true', all search fields have to be matched in order to get a search match.									
send_confirm*	Sends a confirm message to this selected Email.	(confirm.tpl)								
<table><tr><td colspan="2">Input Variables:</td></tr><tr><td>utoken</td><td>This is the user's utoken.</td></tr><tr><td>config.msg</td><td>The body of the confirm Email message that will be sent.</td></tr><tr><td>subject</td><td>The subject of the confirm Email message. (Optional)</td></tr></table>			Input Variables:		utoken	This is the user's utoken.	config.msg	The body of the confirm Email message that will be sent.	subject	The subject of the confirm Email message. (Optional)
Input Variables:										
utoken	This is the user's utoken.									
config.msg	The body of the confirm Email message that will be sent.									
subject	The subject of the confirm Email message. (Optional)									
cmd=test	There are no other parameters, this does a basic test in order to try and locate some common problems.	(internal)								
remove_setting=_xxx	This will remove the named setting from the user's user.dat file (must be a setting beginning with a '_'. can be used as a comma separated list (eg. remove_setting=_sound_file,_real_name)	(selected page)								
page=xxx.tpl	This will allow the administrator to select any template page and display it. This includes any new templates which are desired.	(selected page)								
process=cmd	This will allow the CGI to process another command directly after the main command. (ie cmd=xxx, or button pressed, etc.) This will only get processed if no error occurs.	(command page)								
on_error=cmd	This will allow the CGI to process another command directly after the main command if the main command generates an error. ie. ?cmd=login&user=username&pass=password &on_error=url&link=http://netwinsite.com&utoken=... This can also be used in conjunction with process. In this case, if either the main command or the 'process' command generates an error this command is processed.	(command page)								

force_connection=true	This will force the CGI to set up the network connection to the mail server.	(selected page)
require_lock=true	This will force the CGI to use the file locking routines rather than have the CGI determine whether it should use file locking depending on the command.	(Selected page)
do_admin_fn=true	This will force the CGI to process the auto-delete/auto-logout functions that generally only occur every hour.	(Selected page)
disable_internal_connection=true	Disables 'non-essential' internal connections set up by WebMail	(selected page)

Any of the above commands which have an '*' next to them must be after a number, where this number is normally the message unique ID. The number appears directly after the cmd name.

eg. Cmd=item-1, cmd=item-32, cmd=itempart-5

In most cases, the CGI will automatically take care of building links that require this sort of ID as well. The variables which are displayed in brackets are also required in order to ensure the function behaves correctly.

eg. Cmd=list&fld=INBOX
cmd=moveitem-32&dstfld=New_Box

Every command, with the exception of the register, manager and login commands, require the utoken to be present.

If an error such as 'Incorrect utoken' appears, you should ensure that a user token is being passed, either as a hidden field or as part of a query.

Template Commands

Throughout the templates you will see '||commands||'. These are what the CGI will pick up and replace with data. Some commands set up variables and are only available on one page. Other variables are available on all templates.

There are also conditional statements and functions available. The following tables show all of the available '||commands||'. In the tables, some commands have extra fields, in which case the name of the field will show one of the following:

- 'var' - a variable name.
- 'value' - either a variable or text.
- 'text' - just text.

The below template commands are available on every template:

Table of Contents

- [Conditional Statements](#)
- [Template Commands](#)
- [Extended Template Commands](#)
- [Begin....End Commands](#)

Tables

Conditional Statements		
The below conditional statements are available on every template.		
<pre> ifndef sound Show Picture instead... else ifequal sound loud ...play loud sound file... else ifequal sound normal Play sound file... endif endif endif </pre>		
ifdef var	Conditional inclusion if variable is defined.	
ifndef var	Conditional inclusion if variable is not defined.	
ifinstr value1 value2	Conditional inclusion if template variable value1 contains string	

	value2, case sensitive.
<code> iflower value1 value2 </code>	This will return to true if 'value1' is lower than 'value2'.
<code> ifequal value1 value2 </code>	Conditional inclusion if template value1 equals value2, case insensitive.
<code> ifnequal value1 value2 </code>	Conditional inclusion if template value1 not equals value2, case insensitive.
<code> ifgreater value1 value2 </code>	This will return to true if 'value1' is greater than 'value2'.
<code> else </code>	Optional <code> else </code> section to go with any of the <code> if... </code> conditions.
<code> endif </code>	Marks end of any <code> if... </code> or <code> else </code> section. Must have an <code> endif </code> for every <code> if... </code> .

Template Commands	
<code> define var value </code>	This will create a variable with the stated value.
<code> undef var </code>	This will undefine a variable.
<code> include template </code>	This reads the named file, which may include template variables. ie. <code> include menubar.tpl </code>
<code> do(...) </code>	This is used to run other command line scripts, including CGI's, scripts and other compiled code. NOTE: This will remove the 'Content-Type:' of other CGI's. Variables may also be used as a parameter. ie. <code> do(program.cgi current_user current_host) </code> NOTE: This requires the "do_base_dir" ini setting to be used.
<code> build(...) </code>	This allows the building of variables from variables in order to display their value. <code> define first nw </code> <code> define second img </code> <code> build(first second) </code> or <code> build(NW second) </code> The above will display the value for the variable 'nwimg' if one is available.
<code> lang var </code>	This takes a variable and translates the value using the language file. ie. <code> Define test welcome </code> <code> Lang test </code>
<code> lang_str text </code>	This takes text and translates it using the language file. ie. <code> lang_str INBOX </code>
<code> chop var n </code>	This performs a chop of the variable's value, where it only displays n number of characters.
<code> lchop var n </code>	This will chop the variable to the size provided, chopping at whole word intervals, and adding on '...' if chopped.
<code> chop_str text n </code>	This performs a chop of the text or the name of the variable, where it only displays n number of characters.
<code> lchop_str text n </code>	This will chop the text to the size provided, chopping at whole word intervals, and adding on '...' if chopped.
<code> is_checked var </code>	This will replace with 'checked' if the value of the variable is on, true, checked or a non zero number.
<code> java_text var </code>	This will convert the variable to use '_' for the following characters so that you can use the value as a javascript variable. '(space)', '+', '<', '>', '&', ':'
<code> href_text var </code>	This will convert the variable to use '%xx' for the following characters when using javascript and href's: '(space)', '+', '<', '>', '&', ':' This will only encode the first 1024 characters. This is acceptable in most cases as most browsers have a limit near list on all queries (href's).
	This will convert the variable into displayable HTML characters. This ensures that it will not be treated as an actual HTML. It converts the

html_text var	following characters: ''', '<', '>', '&', '\n', '\r' also language support etc.
html_line var	This works as above, but will not remove the '\n' and '\r's.
wml_text var	This will convert the variable so that it displays correctly in WML.
wml_line var	This works as above, except that it will not remove '\n' and '\r's
wml_line text	This will convert the text so that it displays correctly in WML.
wml_br_text var	This works the same as the variable 'wml_text', except that this variable might have WML code and so is left alone.
wml_br_line var	This works as above, except that it will not remove '\n' and '\r's
wml_str text	This will convert the text so that it displays correctly in WML.
date_today value	This displays today's date in the specified layout, or using the variable provided. ie. date_today date_layout
add var1 var2	Adds 2 integers and displays the result.
sub var1 var2	Subtracts the integer var2 from integer var1 and displays the result.
remove_email var1 var2	Template function, used to remove a particular email address out of a list. Used: remove_email list_of_emails email_to_remove
just_include variable	Like the include template function however, using this command the file will not be phased through the template phaser, it will be displayed as is.

Extended Template Commands											
The below template commands are available on every template.											
/* ... */	Anything between these tags will not be passed out to the web server. This is so that internal comments can be setup that the user cannot see.										
show_email	This will show the selected Email that is setup when calling the item* command.										
ext_display value1	This command takes the variable and encodes it to ensure that it can be correctly sent. This is used for the global external address book.										
folder	This is the unchanged folder name with spaces.										
fld	This is the folder name, encoded to ensure that there are no spaces.										
last_page _current_page	These are setup with the template name so that you can tell what this template is named as well as the last template that was sent out.										
hostlist	This is setup to display the list of available hosts. To have this available on a template you must also send: force_connection=true Layout: <option value=0> hostlist </option>										
fldlist xfldlist	This is set up to display the list of available folders. To have this available on a template you must also send: force_connection=true The xfldlist doesn't include the INBOX folder.										
included_file	Used to indicate how many includes the current point is deep. e.g. displayed template=0, file(s) included in that template=1, file(s) included in any of the originally included file(s)=2, etc, up to a maximum of 5.										
current_user current_host current_port	This is the information about the user, and which host they are connected to.										
folder_stats label	This generates the folder stats for the value of the label provided.										
<table><tr><td colspan="2">Output Variables:</td></tr><tr><td>fld_msg</td><td>This is the number of messages in the folder.</td></tr><tr><td>fld_size/fld_ksize</td><td>This is the size of all the messages in the folder.</td></tr><tr><td>fld_draft</td><td>This is the number of draft messages.</td></tr><tr><td>fld_delete</td><td>This is the number of messages which are marked to be deleted.</td></tr></table>		Output Variables:		fld_msg	This is the number of messages in the folder.	fld_size/fld_ksize	This is the size of all the messages in the folder.	fld_draft	This is the number of draft messages.	fld_delete	This is the number of messages which are marked to be deleted.
Output Variables:											
fld_msg	This is the number of messages in the folder.										
fld_size/fld_ksize	This is the size of all the messages in the folder.										
fld_draft	This is the number of draft messages.										
fld_delete	This is the number of messages which are marked to be deleted.										

	fld_unseen	This is the number of messages which have been read.
	fld_seen	This is the number of unread messages.
	fld_flagged	This is the number of messages that have been flagged.
	fld_reply	This is the number of messages that have been replied to.
length var		This returns the length of the variable's value.
force_sort value		This forces the sorting of the list of Emails. force_sort new force_sort normal
cvt_date var1 var2		This will convert the date field 'var1' to the layout of 'var2'. ie. cvt_date h_date date_layout
divide lot var		This will take the variable value (var) and convert it into separate images for each letter, using the 'lot' as part of the src="...". ie. divide (nwimg /fonts/russian_) h_subject builds: where the 'xxx' is the decimal number of the character. ' ' = 032...etc. When used in conjunction with ' email_charset ' to select which character sets to display, you can set up the page to display in one character set but display other selected character sets using images NOTE: This will not work correctly with characters sets that use wide characters.
disk_quota		The user's disk quota when using POP in bytes.
kdisk_quota		The user's disk quota when using POP on kilobytes.
pop_size		The amount of disk quota that has been used by a user, in bytes.
kpop_size		The amount of disk quota that has been used by a user, in kilobytes.
version		The version number, eg: "v3.0c"
product		The product name ("WebMail")
number_fetch		The number of fetch accounts setup
env value		This returns the environment variable of the value

Begin...End Commands

Most of the following commands require some variables to be setup in order for them to be used correctly. All 'begin...end' commands set up various variables inside them, and normally generate multiple results. The main example is the list begin..end command, where it is displayed for every message on the page.

Any variable in the input list that starts with a '~' is an ini setting.

begin_flag ... end_flag		This is used to display the extended flags of emails.
Input Variables:		
email_extra_flags	This is the list of extra flags available. (Setup by 'begin_list')	
Output Variables:		
flag_name	The name of the flag (ie flag_test)	
flag_short_name	The shorten name of the flag (ie test)	
flag_value	The value of the flag	
begin_list ... end_list		This lists each Email in the selected folder.

Input Variables:	
fld/folder	This is the folder that is selected to display
max_line_count	This is used to setup the available range for the 'line' variable, and to allow highlighting of every other line. (Optional)
_max_message_display ~max_wml_display	This is used to determine how many messages are displayed per page, in HTML or WML mode. (Optional)
pos	This is the starting position of the first Email. It is the actual Email number, not the page number. (Optional)
force_connection=true	This setting is needed to be passed to the CGI if the command is not 'list', 'mail_check', 'mail_reload' and 'fld_manage'.

Output Variables:	
email_charset	This is setup with the character set that the Email states. This can then be used to inform the browser what character set to use. ifdef email_charset <meta http-equiv="Content-type" content="text/html; charset= email_charset "> endif
line	This is setup to go from 0 to 'max_line_count'-1 in values, and to determine how the row is to be highlighted.
msg_no	The message number.
h_uidl/uidl	The UIDL of the Email message.
fet_user* fet_host* fet_port* fet_number*	These are only setup if the user has multi-fetch setup, in which case the values are setup depending on the username, host, port, and what number was setup. Normally the number is used to determine which image to display.
h_isread h_isreplied h_isflagged h_isdeleted h_isdraft h_isunseen h_isremote h_isattached	These are the different flag settings for Email messages. The values of these are '1' or '0', where a '1' indicates that it is active. ie. For 'h_isread', being set to '1' indicates that it has been read.
h_attach	This is the number of attachments that the Email has. On a POP server this is either '0' or '1+'.
h_lines h_bytes/h_kbytes	This is the number of lines, or the size of the Email message. (Not the size of the attachments)
h_to h_cc	This is the list of Email addresses that were sent with this Email message. Note: BCC field is not available.
h_from h_from_email h_from_personal h_reply h_reply_email h_reply_personal	This indicates who the Email came from and who to reply to. This is where you can indicate whether the address should appear in personal, Email or complete mode. ie. Complete: "Lynden" <lynden@1.2.3.4> Email: lynden@1.2.3.4 Personal: "Lynden"
h_subject	This is the subject of the Email.
h_date h_local_date h_date_day h_date_time	This is the date on which the Email was sent. There are various ways to display the date. The last three are all displayed in the timezone setup by the user or administration.
h_pri	This is the priority of the Email. This defaults to 'Normal' if not defined.
b_item b_forward b_forward_edit b_save b_reply b_replyall b_list delitem b_prev/msg_prev b_next/msg_next	These variables are the links to process the stated action. ie. b_forward_edit - is the complete URL to forward_edit the item.

Variables Available after Command:	
list_email_charset	This is set to the character set that should be used to display the list correctly. This can then be used to inform the browser what character set to use. ifdef email_charset <meta http-equiv="Content-type" content="text/html; charset= email_charset "> endif

begin_new ... begin_new	This lists each Email in the selected folder that is marked as new.
---------------------------	---

Input Variables:

This 'begin...end' command has the same input variables as the ' begin_list ... end_list ' command.	
Output Variables:	
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

begin_list_all ... end_list_all	This is used to display the complete list of Email messages.
-----------------------------------	--

Input Variables:	
This 'begin...end' command has the same input variables as the ' begin_list ... end_list ' command.	
Output Variables:	
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

begin_rules ... end_rules	These are the Email filter rules.
-----------------------------	-----------------------------------

Output Variables:	
n	The rule number.
r_headers	The headers with the selected header.
r_contains	The string to be searched for.
r_actions	The selected action.
r_addr	The Email address which should be used, if needed.
r_iscase	This determines whether to use the case dependence
xfldlist	The folder to move/copy items into, if needed.

begin_address ... end_address	This lists the items in the selected address book.
---------------------------------	--

Input Variables:	
abook_edit	The selected address book
Output Variables:	
nick_name	This is the link used to display/edit the data item .
script_nick	This is used to call the javascript: pressnick('nick_name') which is used on the popup address book
name	This is just the nickname.
abk_*	These are all the rest of the address book fields that have been added. The administration sets these up to be any number. ie. abk_email

begin_addr_list ... end_addr_list	This lists the items in the selected address book.
-------------------------------------	--

Input Variables:	
abook_edit	The selected address book
addr_pos	The starting position.
max_wml_display _max_message_display	This is used to determine how many messages to display on a page.
Output Variables:	
nick_name	This is the link used to display/edit the data item.
script_nick	This is used to call the javascript: pressnick('nick_name') which is used on the popup address book
name	This is just the nickname.
abk_*	These are all the rest of the address book fields that have been added. The administration sets these up to be any number. ie. abk_email

begin_fld_short ... end_fld_short	This generates a list of the folders that are available quickly. Excluding the INBOX.
-------------------------------------	---

Input Variables:	
new_mail_name	The name of the INBOX.

inbox_folder	The starting position.
default_folders_first	Weither the default folders are first
user_trash_can	The name of the trash can.
disallow_folders	The folders not to display.
fixed_folders	The list of fixed folders.
remove_fixed_folders	To remove all fixed folders from list.

Output Variables:

fixed	Weither the folder is a fixed folder
fld_name fld_name_short folder_name	The name of the folder. 'folder_name' is already href encoded.
fld_is_public	If the folder is a public folder
fld_no_select	If the folder is not allowed to be selected
fld_prefix_name fld_sub_count	This is used to display the folders into sub folders.

||begin_xfld_short||...||end_xfld_short|| This generates a list of the folders that are available quickly. Including the INBOX.

Input Variables:

This 'begin...end' command has the same input variables as the '||begin_fld_short||...||end_fld_short||' command.

Output Variables:

This 'begin...end' command has the same output variables as the '||begin_fld_short||...||end_fld_short||' command.

||begin_folder||...||end_folder|| This lists the available folders.

Input Variables:

~new_mail_name	This determines what the INBOX should display as. <i>(Optional)</i>
~disallow_folders	This determines which folders are not allowed. <i>(Optional)</i>
~fixed_folders	This determines which folders the users are not allowed to edit. <i>(Optional)</i>
force_connection=true	This setting is needed to be passed to the CGI if the command is not 'list', 'mail_check', 'mail_reload' and 'fld_manage'.

Output Variables:

fld_name	This is the folder's name.
folder_name	This is the folder's name encoded without any spaces. Needed for javascripts or href's
fld_msg	This is the number of messages in the folder.
fld_size/fld_ksize	This is the size of all the messages in the folder.
fld_draft	This is the number of draft messages.
fld_delete	This is the number of messages which are marked to be deleted.
fld_unseen	This is the number of messages which have been read.
fld_seen	This is the number of unread messages.
fld_flagged	This is the number of messages that have been flagged.
fld_reply	This is the number of messages that have been replied to.
fld_fetch	Set to true when INBOX folder is being displayed.

Variables Available after Command:

fld_count	The number of folders.
fld_msg_total	The total number of folders.
fld_size_total fld_ksize_total	The total size of all the folders.
fld_draft_total	The total number of draft messages
fld_delete_total	The total number of messages marked to be deleted .
fld_seen_total	The total number of read messages.
fld_unseen_total	The total number of unread messages.

fld_reply_total	The total number of messages which have been replied to.
fld_tsize_total	The total size of all messages which are not in the trash folder.
fld_ksize_total	
fld_trash_total	The size of the Trash folder.
fld_ktrash_total	
fld_fetch_msg	The number of Emails in the fetch accounts.
fld_fetch_size	The size of all the mail in bytes in the fetch accounts.
fld_fetch_ksize	The size in Kbytes
fld_fetch_draft	The number of draft messages in fetch accounts.
fld_fetch_delete	The number of marked to be deleted messages in fetch accounts.
fld_fetch_seen	The number of seen messages in fetch accounts.
fld_fetch_unseen	The number of unseen messages in fetch accounts.
fld_fetch_flagged	The number of flagged messages in fetch accounts.
fld_fetch_reply	The number of replied messages in fetch accounts.

||begin_fld_list||...||end_fld_list|| This lists the available folders, only displaying a limited number per page.

Input Variables:	
~new_mail_name	This determines what the INBOX should display as. <i>(Optional)</i>
~disallow_folders	This determines which folders are not allowed. <i>(Optional)</i>
~fixed_folders	This determines which folders the users are not allowed to edit. <i>(Optional)</i>
force_connection=true	This setting is needed to be passed to the CGI if the command is not 'list', 'mail_check', 'mail_reload' and 'fld_manage'.
Output Variables:	
fld_name	This is the folder's name.
folder_name	This is the folder's name encoded without any spaces. Needed for javascripts or href's
fld_msg	This is the number of messages in the folder.
fld_size/fld_ksize	This is the size of all the messages in the folder.
fld_draft	This is the number of draft messages.
fld_delete	This is the number of messages which are marked to be deleted.
fld_unseen	This is the number of messages which have been read.
fld_seen	This is the number of unread messages.
fld_flagged	This is the number of messages that have been flagged.
fld_reply	This is the number of messages that have been replied to.
fld_fetch	Set to true when INBOX folder is being displayed.
Variables Available after Command:	
fld_count	The number of folders.
fld_msg_total	The total number of folders.
fld_size_total	The total size of all the folders.
fld_ksize_total	
fld_draft_total	The total number of draft messages
fld_delete_total	The total number of messages marked to be deleted .
fld_seen_total	The total number of read messages.
fld_unseen_total	The total number of unread messages.
fld_reply_total	The total number of messages which have been replied to.
fld_tsize_total	The total size of all messages which are not in the trash folder.
fld_ksize_total	
fld_trash_total	The size of the Trash folder.
fld_ktrash_total	
fld_fetch_msg	The number of Emails in the fetch accounts.
fld_fetch_size	The size of all the mail in bytes in the fetch accounts.
fld_fetch_ksize	The size in Kbytes

fld_fetch_draft	The number of draft messages in fetch accounts.
fld_fetch_delete	The number of marked to be deleted messages in fetch accounts.
fld_fetch_seen	The number of seen messages in fetch accounts.
fld_fetch_unseen	The number of unseen messages in fetch accounts.
fld_fetch_flagged	The number of flagged messages in fetch accounts.
fld_fetch_reply	The number of replied messages in fetch accounts.

||begin_bulletin||...||end_bulletin|| This lists the available address books.

Input Variables:

~bulletin_path	This is where the bulletins are stored. This MUST be setup.
max_line_count	This is used to setup the available range for the 'line' variable. This is used to allow highlighting of every other line. <i>(Optional)</i>

Output Variables:

new_bulletin	This is set to 'yes' if the bulletin is new to the user.
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

||begin_forward||...||end_forward|| This lists the Emails that will be forwarded.

Input Variables:

fwd_list	This is the list of UIDL's of the Emails which will be forwarded.
Folder	This is the folder to which the Items will be forwarded.

Output Variables:

This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	
--	--

||begin_users||...||end_users|| This lists the result of the search for users.
(Used only on manager's page)

Input Variables:

wild_search	The wild search string.
-------------	-------------------------

Output Variables:

user	The user's complete user directory.
user_name	The user's username.
user_host	The user's host name.
hash_type	The hashing method which is being used. (1 or 2)

||begin_pages||...||end_pages|| This generates the list of page numbers.

Input Variables:

num_pages	The number of pages.
_max_message_display ~max_wml_display	The number of items per page.

Output Variables:

npage	The page number.
pagepos	The position of the page number using the number of items per page.
pagestart	The position of the start of this page. (ie pagepos)
pagefinsih	The position of the end of this page.

||begin_xpages||x||...||end_xpages|| This generates the list of page numbers and displays no more than x at any one time.

Input Variables:

num_pos	The current page number
num_pages	The number of pages.

_max_message_display ~max_wml_display	The number of items per page.
Output Variables:	
npage	The page number.
pagepos	The position of the page number using the number of items per page.

||begin_todo||...||end_todo||

This lists the items on your 'to do' list.

Input Variables:	
~timezone	The timezone which the times will display in.
Output Variables:	
tdl_name	The name/header of the 'to do' list.
tdl_*	The list of administration settings. ie. tdl_data
tdl_old	This is set to 'true' if the item is old.
tdl_today	This is set to 'true' if the item is from today.
tdl_date	The date which the 'to do' list has been set to.
tdl_std_date	The date at GMT time.

||begin_profile||...||end_profile||

Lists the user's profiles.

Output Variables:	
profile_name	This is the name of the profile.
pro_*	This is the list of administration settings. ie. pro_sig

||begin_tpl||...||end_tpl||

This lists the available templates.

Input Variables:	
~tpl_set	This ini setting lists each template set available to the user.
Output Variables:	
tpl_number	The template number.
tpl_path	The path to the template.
tpl_text	The name of the template set.

||begin_search||...||end_search||

This command displays the results of your Email search.

Input Variables:	
max_line_count	This is used to setup the available range for the 'line' variable. This is used to allow highlighting of every other line. <i>(Optional)</i>
search_mode	When set, the search mode is OR.
search_from	The 'from' seach string.
search_subject	The 'subject' search string.
search_body	The 'body' search string.
search_body_size	The maximum number of characters in the body which the search command will check against.
multi_sel_fld	The list of folders which the search function will look in.
force_connection=true	This setting is needed to be passed to the CGI if the command is not 'list', 'mail_check', 'mail_reload' and 'fld_manage'.
Output Variables:	
h_search_from	The converted search string.
h_search_subject	The converted search string.

h_search_body	The converted search string.
h_folder	The folder where the Email message is located.
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

||begin_search_list||...||end_search_list|| This command displays the results of your Email search.

Input Variables:	
max_line_count	This is used to setup the available range for the 'line' variable. This is used to allow highlighting of every other line. <i>(Optional)</i>
search_mode	When set, the search mode is OR.
search_from	The 'from' seach string.
search_subject	The 'subject' search string.
search_body	The 'body' search string.
search_body_size	The maximum number of characters in the body which the search command will check against.
multi_sel_fld	The list of folders which the search function will look in.
force_connection=true	This setting is needed to be passed to the CGI if the command is not 'list', 'mail_check', 'mail_reload' and 'fld_manage'.
Output Variables:	
h_search_from	The converted search string.
h_search_subject	The converted search string.
h_search_body	The converted search string.
h_folder	The folder where the Email message is located.
This 'begin...end' command has the same output variables as the ' begin_list ... end_list ' command.	

||begin_timezone||...||end_timezone|| Lists the timezone settings which are available.

Input Variables:	
~timezone	The timezone which the user has currently selected.
Output Variables:	
tz	The timezone letters. (ie. nzst)
tz_name	The full timezone name. (ie New Zealand Standard)
tz_offset	The timezone offset. (ie +1200)
tz_selected	This is set if the timezone has been selected.

||begin_ext_book||...||end_ext_book|| The list of global external address books.

Input Variables:	
ext_books	This lists the available books, and is setup internally.
Output Variables:	
book_name	The name of the book
book_text	The general text about the book.
book_edit	When set to 'true' the user can change the addresses in this address book.

||begin_ext_avail||...||end_ext_avail|| The list of available variables to display.

Input Variables:	
ext_available	The list of available fields. This is setup internally.
Output Variables:	

field_name	The name of the field. (ie phone_number)
field_text	The name in general readable text. (ie Phone Number)

||begin_ext_search||...||begin_ext_search||The list of available fields which may be searched on.

Input Variables:	
ext_search	The list of available search fields. This is setup internally.+++
Output Variables:	
field_name	The name of the field. (ie phone_number)
field_text	The name in general readable text. (ie Phone Number)
field_type	The type of field. (string, number, date etc..)

||begin_ext_results||...||end_ext_results||The list of the results from the search.

Input Variables:	
ext_results	The list of search results. This is setup internally.
Output Variables:	
ext_result	The resulting line.

||begin_attach||...||end_attach||The list of attached files.

Output Variables:	
attach_fname	The complete filename.
attach_name	The name of the file, excluding the extention.
attach_size/attach_ksize	The size of the file.
Variables Available after Command:	
attach_total_size attach_total_ksize	The complete size of all files.

||begin_keep_attach||...||end_keep_attach||The list of files which are stored by the CGI.

Output Variables:	
attach_fname	The complete filename.
attach_name	The name of the file, excluding the extention.
attach_size/attach_ksize	The size of the file.
Variables Available after Command:	
attach_total_size attach_total_ksize	The complete size of all files.

||begin_pick||...||end_pick||The Emails which are in the pick list.

Input Variables:	
pick_item	The list of available fields. This is setup internally.
Output Variables:	
pick_name pick_email pick_personal	The Email of the pick item. This is where you can display whether the address appears in personal, Email or complete format. ie. Complete: "Lynden" <lynden@1.2.3.4> Email: lynden@1.2.3.4 Personal: "Lynden"

||begin_fet||...||end_fet||The list of all the multi-fetch accounts.

Input Variables:	
host_fetch	The list of available fields. This is setup internally.
Output Variables:	

fet_count	The count of the fetch.
fet_user	The username of the fetch account.
fet_host	The host of the fetch account.
fet_port	The port of the fetch account. (ie 110, 143)
fet_number	This is just an extra number that can be stored. Normally used to determine which image to display.
fet_active	This is set if the account is active.

begin_sel ... end_sel	Cycle though all the 'Sel_*' fields listing each one.
-------------------------	---

Input Variables:	
sel_*	The list of selected boxes.
Output Variables:	
sel	The name of the select.

begin_host ... end_host	The list all the current setup hosts lines.
---------------------------	---

Output Variables:	
cur_num	The host number.
cur_host	The host name/IP.
cur_user	The username setup for this host.
cur_port	The port of the host account. (ie 110, 143)
cur_prefix	The IMAP prefix that will be used with this host.
cur_profile	The selected profile to be used with this host.

begin_for x y ... end_for	This cycles through starting at 'x' and finishing at 'y'.
-------------------------------	---

Output Variables:	
for	The count it is on.

begin_email email ... end_email	This will break up the email list provided into seperate fields.
------------------------------------	--

Input Variables:	
email	A comma seperated email list to be broken up.
Output Variables:	
num	Number.
email email_email email_personal	The full email address, just the email and just the personal details.

begin_email email ... end_email	This will break up the email list provided into seperate fields.
------------------------------------	--

Input Variables:	
email	A comma seperated email list to be broken up.
Output Variables:	
num	Number.
email email_email email_personal	The full email address, just the email and just the personal details.

Multiple Template Setup

WebMail supports multiple template sets, allowing different styles or even different languages on the one site. Each template can have it's own template directory as well as it's own image location. In your ini file, add one line for each template set that you wish to have, in the following layout:

```
tpl_set <number> <templates_dir> <nwimg> <name>
```

The 'number' is the template number that is used to determine which template you are using, this should be unique. The 'templates_dir' is the full path to the template directory.
The 'nwimg' is the web server relative path to the images.
And 'name' is just text that is displayed to the user about the template set.

Example:

```
tpl_set 1 /var/spool/webmail/marble /nwimg/mail/marble Marble
tpl_set 2 /var/spool/webmail/iconic /nwimg/mail/iconic Iconic
tpl_set 3 /var/spool/webmail/globe /nwimg/mail/globe Globe
```

The next step is changing the templates to allow the users to select which templates they want to use. Below are the template additions needed in order to allow the user to select which template to use.

login.tpl

```
||ifdef||multiple_tpl||
<!-- If you wish multiple template sets to be user selected then remove the following --->
<!-- The users can then select their template sets on the configuration page. --->
<tr>
<td align="right">Select Template Set: </td>
<td> <select name="selected_tpl" size=1>
<option value="">(Default)</option>
||begin_tpl||
<option value="||tpl_number||">||tpl_text||</option>
||end_tpl||
</select>
</td>
</tr>
||endif||
```

config.tpl

```
||ifdef||multiple_tpl||
<tr>
<td align="right">Select Template Set: </td>
<td> <select name="_selected_tpl" size=1>
||begin_tpl||
||ifequal||_selected_tpl||tpl_number||
<option selected value="||tpl_number||">||tpl_text||</option>
||else||
<option value="||tpl_number||">||tpl_text||</option>
||endif||
||end_tpl||
</select>
</td>
</tr>
||endif||
```

SMSGate: SMS to Email gateway

It allows you to provide SMS message facilities via email using a gsm modem. It has settable limits for email address, SMS number and IP's, plus complete logs of traffic for billing purposes.

It can be integrated into any mail server allowing sms notification of email, or simply the ability to send sms via thier existing email client. It can also be integrated into our DBabble instant message server to allow instant sms messages to be sent.

The following sections describe...

- 1. [NT Installation.](#)
- 2. [Unix Installation.](#)
- 3. [Important Settings](#)
- 4. [General usage.](#)
- 5. [Commands](#)
- 6. [Configuration settings.](#)
- 7. [Delivery Modes](#)
- 8. [Troubleshooting](#)

NT Installation

Download the self extracting archive from [here](#), run it and follow the on screen instructions. Or if you would like to perform an install manuall you can follow the steps below.

- Extract the self-extracting archive to install directory.
- Run msgate -install
- Copy msgate.ini to the windows directory, i.e. c:\windows or c:\winnt and edit the settings. see [here](#)
- Type "net start msgate"

Unix Installation

Download the self extracting archive from [here](#), run it and follow the on screen instructions. Or if you would like to perform an install manuall you can follow the steps below.

- Extract the tarball into install directory.
- Run ./msgate -install.
- Copy msgate.ini to the /etc directory, and edit the settings. see [here](#).
- Add SMSGate to your startup scripts.

Important Settings

Setting	Default	Description
smtp_inport	1025	Port to listen on and accept incomming SMTP messages
smtp_outserver	<hostname>	Ip address/name of the normal SMTP server for outgoing email messages.
smtp_outport	25	Port the normal SMTP server is listening on.
smtp_prefix		Prefix to strip from incoming email addresses, and to add onto outgoing email addresses, to turn them into/from phone numbers.
smtp_domain	<hostname>	Domain name to add to outgoing phone numbers to make them a correct email address.
delivery_mode	gsm	Method of SMS delivery, currently only 'GSM' is supported, plans to add SMPP, SMTP, and XML - Let us know if you require a different transport method from those currently supported.
phone_country		Country code for local phones, i.e. 64 is New Zealand
phone_service		Area / Service code for local phones i.e. 025 for Telecom, 021 for Vodaphone (New Zealand values shown)
gsm_com	COM1	Com port the GSM modem is connected to. For ports above COM9 the setting must be formatted \\.\COMxx where xx is the port number, eg \\.\COM40
gsm_baud	9600	Baud rate to use talking to modem.
gsm_number		GSM modem phone number.
workpath	/msgate	Path to store work files, traffic logs etc.
logpath	<workpath>	Path to store debug log files.
loglevel	info	Level of debugging.. none, error, info, and debug are valid.
accept_ip	*	Wildcard list of IP address to accept smtp messages from.

from_limit		Maximum number of messages from a particular email address in a single day.
to_limit		Maximum number of messages to a particular phone number in a single day.
ip_limit		Maximum number of messages from a particular IPin a single day.

General Usage

SMSGate is a server, it accepts SMTP style connections on a specified port and delivers those messages via SMS messages to the given SMS phone number. It also recieves SMS messages and delivers them via email to either:

- The address specified in the SMS message.
- The email address that sent an SMS thru SMSGate to that phone number (if there is more than 1 it bounces an SMS asking for confirmation of the email address desired).

It is expected that you will run SMSGate in conjunction with a normal SMTP server and gateway specific messages thru to it, allowing SMS notification of email and other useful features.

A normal SMTP server is NOT required for incoming SMTP messages but IS required for outgoing email messages.

SMSGate can run in conjunction with DBabble allowing you to give sms sending capabilities to your DBabble instant message users.

Configuration settings

A default enclosed in <> means the setting defaults to the setting name in the <>. <hostname> is a special case, the hostname is the computers host name and it is looked up upon execution. A setting with 'multi' in it's description can have multiple values.

Setting	Default	Description
accept_from	*	multi; Wildcard list of email addresses to accept smtp messages from.
accept_ip	*	multi; Wildcard list of IP address to accept smtp messages from.
accept_to	*	multi; Wildcard list of phone numbers to send sms to.
alias_file		see below
cbst_one	false	Deprecated; This is now configured automatically.
cnmi_flush	false	Controls how incoming message indications are flushed, do not change unless troubleshooting.
com_debug_file		The file in which to log data to/from the GSM modem.
cpin_ok	false	Tells SMSGate to expect an extra OK message from the CPIN command.
delivery_file		see below
delivery_mode	gsm	Method of SMS delivery, currently only 'GSM' is supported, plans to add SMPP, SMTP, and XML - Let us know if you require a different transport method from those currently supported.
email_subject	\$(BODY20)	Specifies the format of the subject in the email generated from an incoming SMS. May contain \$(xy) where x can be from, body, or any smsgate.ini configuration setting and y is the length to truncate that value to. eg. \$(from5) \$(body10)
from_limit	500	Maximum number of messages from a particular email address in a single day.
from_limit_msg	<limit_msg>	The message given when the from_limit is reached.
gsm_baud	9600	Baud rate to use talking to modem.
gsm_com	COM1	Com port the GSM modem is connected to.
gsm_crlf	true	Deprecated; This is now configured automatically.
gsm_init	AT	The initialisation string sent to the GSM modem.
gsm_min_signal	0	If set > 0 SMSGate waits till the modem signal strength is at least this value. Possible values include 0 thru 31, 31 being the best possible signal strength.
gsm_noinit	false	Disables the gsm_init string.
gsm_number		GSM modem phone number.
gsm_pin		The PIN number required to access the GSM modem.
gsm_pin2		The PIN2 number required to access the GSM modem.
gsm_puk		The PUK required to re-set the PIN number.
gsm_puk2		The PUK2 required to re-set the PIN2 number.
gsm_signal_timeout	60	How long to wait (secs) for gsm_min_signal level to be reached.
home	\\smsgate	The directory to read and write smsgate files from/to.
host	<hostname>	Automatically configured to the hostname of the computer.

ip_limit	0	Maximum number of messages from a particular IP in a single day.
ip_limit_msg	<limit_msg>	The message to give when ip_limit is reached.
limit		multi; see below
limit_msg	Limit reached	The default message to give when a limit is reached.
log_delimiter		This is the character to use to delimit fields in the logs, setting it to "TAB" causes the tab character to be used.
loglevel	debug	Level of debugging.. none, error, info, and debug are valid.
logpath	<workpath>	Path to store debug log files.
phone_country		Country code for local phones, i.e. 64 is New Zealand
phone_range		The range of phone numbers assigned to your SMPP connection, only used by delivery_mode smpp. (which is completed, tested, but has not been used in a real live enviroment).
phone_service		Area / Service code for local phones i.e. 025 for Telecom, 021 for Vodaphone (New Zealand values shown)
recv_mode	<delivery_mode>	Specifies the mode to use to receive SMS messages. Set this if it's different to delivery_mode.
reply_with	true	Causes the text "Reply w/ EMAIL" to be prepended to outgoing sms messages.
report		multi; see below
robot_cmd		If using deliver_mode robot then this specifies the robot, this command is executed and the sms to deliver is sent to it on stdin. Other information is passed in enviroment variables: SMS_LENGTH, SMS_FROM, and SMS_TO.
send_mode	<delivery_mode>	Specifies the mode to use to send SMS messages. Set this if it's different to delivery_mode.
smpp_addr_range	*	This is sent to the SMPP server as part of the initial connection, it specifies the addresses you are interested in.
smpp_host		The hostname or ip of the SMPP server to use.
smpp_pass		The password for access to the SMPP server.
smpp_port	2775	The port the SMPP server is listening for connections on.
smpp_user		The username for access to the SMPP server.
sms_best_guess	false	When determining the email to send an incoming SMS to SMSGate will check in the message text for a destination. Failing that it will send to the email it has recorded as sending an sms to the incoming number, provided there is only 1 of them, if there are more than 1 it fails, unless you use this setting, it will cause it to send to the most recent.
sms_bounces	true	When SMSGate cannot determine which email to send the incoming sms to, it generates a bounce or error sms message and sends it back to the sms sender, this disables those bounces.
sms_dcs	7bit	This is the Data Coding Sceme to use in SMS messages, options are 7bit, 8bit, and 16bit. Only applicable in "sms_mode pdu". The larger the bit count the less characters will fit in the message, but the greater the range of characters you will be able to use.
sms_extended	false	This setting enables the extended 7bit DCS mode.
sms_format		This specifies the format of the sms when converting email to sms. It can contain special inserts like \$(bodyX) \$(nl) or \$(headerX) where header can be any email header. The X specifies the length to truncate the value at eg. \$(body10) shows 10 characters from the body of the email. eg. sms_format \$(subject20)\$(nl)\$(body)
sms_ignore		multi; Tells msgate to ignore incoming sms messages from this phone number.
sms_mode	text	SMS messages can be sent in either 'text' or 'pdu' mode, 'text' is human readable, 'pdu' is encoded data.
sms_retries	3	Number of times to retry sending an SMS message.
sms_smart		This is an extensible setting for specifying formatting options within sms message, the only option available currently is "spaces" which causes msgate to strip spaces from text and capitalise the first letter of each word, this is intended to squeeze more text into the SMS message.
sms_static	false	This causes msgate to deliver all incoming SMS messages to the same email address, that address is made up of the smtp_prefix, gsm_number and smtp_domain settings eg. <prefix><number>@<domain>
sms_subject	true	This cause SMSGate to include the email subject in the SMS message body, followed by

		an '-' eg. SUBJECT - BODY
sms_to		multi; see below
smtp_debug_file		This specifies a file to log all SMTP data to.
smtp_domain	<host>	Domain name to add to outgoing phone numbers to make them a correct email address.
smtp_greeting	SMSGate	The initial SMTP greeting string.
smtp_inport	1025	The port on which SMSGate listens for incoming SMTP connections.
smtp_outport	25	Port the normal SMTP server is listening on.
smtp_outserver	<host>	Ip address/name of the normal SMTP server for outgoing email messages.
smtp_prefix		Prefix to strip from incoming email addresses, and to add onto outgoing email addresses, to turn them into/from phone numbers.
smtp_retries	3	Number of times to retry sending an email via the smtp_outserver.
smtp_retry_wait	5	Time (mins) to wait between smtp_retries attempts.
ssl_cert	sg_cert.pem	File containing the SSL certificate
ssl_dir	<workpath>	Directory containing ssl_cert and ssl_priv
ssl_priv	sg_priv.pem	File containing the SSL private key
to_limit	500	Maximum number of messages to a particular phone number in a single day.
to_limit_msg	<limit_msg>	Message to give if to_limit is reached.
web_allow	127.0.0.1	Port to allow incoming web admin requests on.
web_https	true	Web admin is an HTTPS port.
web_port	8775	Port to accept web admin connections on.
webpath	./web	Path to web admin files.
workpath	<home>	Path to store work files, traffic logs etc.

The more complicated settings have several parts to them and require more of an explanation. The format is shown below on several lines for clarity, in the configuration file each setting should be on one line.

Setting	Format	Description
alias_file	file=VALUE delimiter=VALUE order=VALUE required=VALUE	This file is used to translate the phone number of an incoming sms into an email address which is then set as the From address in the email. The delimiter and order fields specify the format of the file (see below). Required specifies that a translation is required, if none is found the message is not sent.
delivery_file	file=VALUE delimiter=VALUE order=VALUE	This file is used to route an incoming sms based on the the phone number it comes from to a destination email address. The delimiter and order fields specify the format of the file (see below).
limit	name=VALUE ip=VALUE to=VALUE from=VALUE period=VALUE	Specifies a rule limiting the SMS messages that can be sent in the specified time period. Similar to ip_limit, to_limit and from_limit except that it allows configuration of the time period and name of the rule. ip_limit, to_limit and from_limit are identical to a limit setting like: limit name="msgrec" ip="x" to="y" from="z" period="24" which is a rule called msgrec which allows x sms from any given ip, y sms to any given number, and z sms from and given email address in a 24 hour period. The period field is in hours, unless a unit is specified (d)ay, (w)week, (y)ear eg. period="2w" means 2 weeks.
report	time=VALUE to=VALUE message=VALUE days=VALUE	Causes SMSGate to send a periodic SMS message to prove it is operational. Time is specified as hour:min eg. 21:45. To is a phone number to send the report to. Message is the text to include in the report. Days is a comma seperated list of days of the week and/or a range eg. Sun,Tue-Thu,Sat
sms_to	keyword=VALUE phone=VALUE prefix=VALUE domain=VALUE	SMSGate looks in the body of all incoming SMS messages for the specified keyword, the sms must be from a number matching the specified 'phone' wildcard. The sms is then sent to the email address constructed from prefix, domain and the word following the keyword. eg: sms_to keyword="dbabble" phone="*" prefix="dbabble_" domain="domain.com" will send an email to 'dbabble_regan@domain.com' if an incoming sms contains "dbabble regan".

The delimiter and order fields of alias_file and delivery_file specify the format of the file. Delimiter may be set to any character or sequence of characters. One special case "TAB" is supplied to set delimiter to the tab character. The order field is a comma seperated list with two possible values PHONE and EMAIL eg. order="PHONE,EMAIL".

For example a setting: alias_file file="aliases.txt" delimiter="|" order="PHONE,EMAIL" required="TRUE" would expect the aliases.txt file to look a bit like this:

+64213334444|regan@netwin.co.nz
+64256667777|regan@netwinsite.com

Delivery modes

SMSGate currently supports several different SMS delivery modes; delivery via a GSM modem connected to the PC via the COM (serial rs232) port (often a USB connection emulating a COM port), delivery via an online service using HTTP or SMTP as the transport method and the facility to integrate your own method via "robot" mode.

In addition SMSGate has untested SMPP support. If you have an existing SMPP connection and are interested in SMSGate we would be pleased to work with you testing SMSGate for use with your connection. It would give us the opportunity to fix any problems and refine any rough edges present in the SMPP code.

To configure these delivery modes you edit the smsgate.ini file which can be found in the c:\windows or /etc directories (note that there is a backup copy in the smsgate installation directory eg c:\smsgate, do not confuse this with the active copy)

To use a GSM modem you configure the delivery_mode (or send_mode and recv_mode) setting(s) to "gsm" and configure the modem using the various gsm_ settings eg. gsm_com, gsm_baud, gsm_number, etc. There are also a number of compatibility settings eg cpin_ok for various modem behaviours we have encountered, you should initially ignore these settings and only modify them when consulting the [Troubleshooting](#) section below.

To use an online service which expects SMTP requests to send SMS you use "smtp" mode. This mode allows you to send SMS but does not allow the receipt of replies. To configure this mode you set send_mode to "smtp" and recv_mode to "none". It is in fact possible to send using SMTP online SMS services without using SMSGate, you could configure SurgeMail to send to the service directly. One reason you might use SMSGate is that SMSGate will parse the incoming email, strip html tags and format the email as specified by the sms_format (and other settings) this parsing can be very useful.

To use an online service which expects HTTP requests to send SMS you use what we call "robot" mode, this allows you to send SMS but does not allow the receipt of replies. To configure robot mode you set send_mode to "robot" and recv_mode to "none". Next you setup the robot process with robot_cmd, eg. robot_cmd c:\smsgate\sms_robot.exe. This sms_robot.exe binary should come with your SMSGate download. Lastly you configure the robot process itself, you do this by editing the sms_robot.ini configuration file. The configuration file comes with some example settings, modify them to suit what the online service expects.

To integrate your own method you use "robot" mode as described above in the handling of HTTP requests. The difference is that you code/provide your own robot process which delivers the sms message in it's own way. The "robot" protocol is fairly simple, SMSGate will execute the specified process and pass the following enviroment variables:

SMS_LENGTH	The length of the formatted SMS message in bytes.
SMS_FROM	The from email address.
SMS_TO	The destination SMS number.

It will then send the body of the formatted SMS message to the process on it's stdin pipe. It expects a single line response saying either "SUCCESS" indicating the message was delivered or a one line status or error message indicating the reason for the failure. This response is logged in the message record logs.

Troubleshooting

If SMSGate will not start, it can be one of several reasons:

- A port cannot be bound, as something else is already using it.
- The GSM Modem could not be contacted on the gsm_com port.
- The GSM Modem did not respond as expected.

To identify what problem SMSGate is having you should have a look in the smsgate.log file, which can be found in the smsgate installation directory (c:\smsgate or /usr/local/smsgate). It will contain the error, look for a line containing "Error:", this will give a clue as to why it will not start.

If the Error line says:

- "Failed to load configuration file..." then smsgate.ini has not been created or is in-accessable (see install.txt)
- "License key faulty..." then there is a problem with the license key, the rest of the error should tell you what.
- "Failed to read (...)" then it is having trouble reading the specified file, does it have permission?
- "Failed to load message rec..." then it is having trouble reading the msgrec.ip,msgrec.from,msgrec.to files.
- "Config setting 'gsm_number' required" then you need to specify gsm_number in smsgate.ini
- "Robot_cmd is required for this delivery_mode..." then you need to specify robot_cmd in smsgate.ini
- "Failed to init command channel..." then the port specified by the web_port setting in smsgate.ini is being used, change this setting to another port.
- "Failed to init smtp..." then the port specified by the smtp_inport setting in smsgate.ini is being used, change this setting to another port.

- "Failed to init gsm..." there is a problem connecting to or initializing the modem, set the `com_debug_file` setting to a path and filename where it can write some debugging information, then try to start it again, now open that debug file and have a look at the last command it carried out, if it was:
 - "AT" then add the setting "`gsm_noinit true`" to `smsgate.ini` and try again.
 - "AT+CBST..." then add the setting "`cbst_one true`" to `smsgate.ini` and try again.
 - "AT+CMGF..." then add the setting "`sms_mode pdu`" to `smsgate.ini` and try again.

If it starts ok, but behaves strangely then look in the `com_raw.log`, and find the AT+CPIN command, if the modem has responded with two lines "+CPIN:.." and then "OK" you need to add the "`cpin_ok true`" setting to `smsgate.ini` and try it again.

A good test is to connect to the modem with Hyperterminal and see if you can communicate with it, experiment with different baud settings etc. Some commands to send as a test AT, ATE1, ATS3?, ATS4?. Remember, if you're trying to connect to a port number higher than COM9 you need the alternate `gsm_com` setting format, eg. `gsm_com \\.\COM40`

If the problem is not described above then please send all the log files created by `smsgate` to smsgate-support@netwinsite.com with a description of the behaviour and we will help you solve it.



DList - Mailing Lists

- [DList - Quick Overview](#)
- [What is a Mailing List?](#)
- [Creating a Mailing List](#)
- [Mailing lists on Virtual Domains](#)
- [Adding Users to a List](#)
- [Settings - lists.dat](#)
- [An example lists.dat file](#)
- [Welcome Messages](#)
- [List Footers](#)
- [Moderated Lists](#)
- [Archives and Files](#)
- [DList Email Commands](#)
- [User's Real Names](#)
- [Template Footers](#)
- [Template Variables](#)
- [Language translation Features](#)

DList - Quick Overview

DList is a mailing list server that is automatically installed as part of SurgeMail

General settings for DList are contained in the main configuration file, dlist.ini.

To create a list you simply add a line like

```
list listname
```

setting in the file [lists.dat](#) which you will find in the DList directory. Then make SurgeMail reload the configuration file (DList regularly checks the configuration and lists.dat files for changes so it does not need to be sent a reload command).

Generally the sysadmin would set up a list and then users would send an email to the 'listname-request' address to 'subscribe' themselves to the list.

Users in general will only interact with the list by sending emails, either directly to the list to be 'posted' or to the listname-request address if they wish to join the list or send it commands. See:

[DList mail commands](#)

When users join the list they are normally sent this list of commands so that they know what the list can do for them.

To modify DList settings you can directly edit dlist.ini and lists.dat with a text editor or use the web admin tool.

Creating a Mailing List - Manually

To create a mailing list on the list server DList, you need to add a new list setting in the lists.dat file, eg:

```
list listname
```

where listname is the name of the list. You can edit the lists.dat file with a text editor (e.g. notepad or vi) or use the web admin tool.

If you are doing it manually then below the list setting add any other settings that you require for your list, eg:

```
title juggling
```

Issue a 'tellmail reload' command so that SurgeMail notices the new list

To try out the list, you should add a user to the list and then post a message to the list. For information on this see,

[Adding Users to a List](#)

[DList Email Commands](#)

Creating a Mailing List - Web Interface.

You can also create mailing lists in the web admin, domain admin or user self admin interfaces. You can define defaults for new mailing lists by creating a file called list_defaults.txt in the web directory, in that file place the defaults

you wish, e.g. to see what variables exist examine na_list.htm, in general prepend 'list_' to the dlist setting. e.g.

```
list_archive true
list_reply_to_user true
```

NB: Mailing lists on Virtual Domains...

If you are wanting to add the mailing list to a specific domain eg: a virtual domain, then you need to specify that domain in lists.dat so that SurgeMail can create the correct aliases for your mailing list.

There are two ways to do this.

1. Old way: add a domain setting to your list eg:

```
list juggling
  title Mailing List
  domain vdomain1.com
```

2. New way: create the list with a full list name, eg:

```
list juggling@vdomain1.com
  title Mailing List
```

The second method is better because it means that all mailing list directoryies will be created with unique names. This allows you to reuse mailing list names on different domains.

Settings - lists.dat

Lists.dat is the file where you create all the lists on your DList server and where you enter individual settings for each list.

All settings are one per line, and you can exclude a line by starting it with the '#' character. You do not need to reload the DList server after making changes.

Below is a list of all of the settings available for each list. All settings for a list are entered on the lines following the

```
list listname
```

line that declares a list, before the next list starts with its 'list listname' declaration line. See [example lists.dat file](#)

All settings take just ONE value except where stated otherwise in the description.

Note: In the table below you will see that the 'access' settings can generally take one of the following values. It is important to think about what these settings mean - NOT all of them apply to every access setting!

- member - refers to list members and in general the list moderator as well
- anyone - no restriction
- moderator - only the list moderator can do it
- *domain - person trying to do it must have the email address ending in 'domain'

Setting	Default	Example/options	Description
allow_ip	*	10.1.2.*	List specific IP addresses that can send to this mailing list, good for securing a large outgoing only list against spam
access_join	anyone	anyone,*netwinsite.com,	Controls who can join the mailing list
access_leave	anyone	moderator	Controls who can unsubscribe from the mailing list, by default anyone can unsubscribe anyone else. in version 2.5d (2.4k) and above: members: (can unsubscribe themselves, moderator can unsubscribe anyone) moderator: (only moderator can unsubscribe - members cannot unsub themselves)
access_post	members	moderator	Controls who can post messages to the mailing list
access_get	members	moderator	Controls who can get messages or files from the archive.

access_who	moderator	members	Controls who can retrieve the list of current members. Note the default changed from members to moderator in SurgeMail 2.3
archive	false	true	If set DList will record all incoming messages in an 'archive' sub directory, off the list's directory.
bounce_remove (added in 2.8h)	false	true	If set then DList will log all bounces that it receives, and if it can work out that the bounce was because of a permanent error, then it will remove that address from the mailing list. DList will also send a summary email to the moderator each day recording any addresses it has removed from the list and the bounce error that was the reason for the removal. This is a new 'beta' setting so let us know how it goes if you try it. You might like to try, the log_bounce setting as a first step to turning on this setting.
domain	(none>	domain mydomain.com	Specifies the domain that this list should exist on where you do not want it to be on your first host_domain. NB: To allow listname re-use on different domains see the note, Mailing lists on Virtual Domains
footer (version 2.4h and above)	(none)	footer c:\surgeemail\dlist \listname\footer.txt	The full path to a file that you want added onto the end of all messages as a footer. In version 2.8e and above this is only added onto all TEXT messages, HTML version also added see below. Note that as of 2.9g, template variables can be used in footers. See Template Footers for details.
footer_html (version 2.8e and above)	(none)	footer_html c:\surgeemail\dlist \listname\footer_html.txt	The full path to a file that you want added onto the end of all HTML messages as a footer. Note that as of 2.9g, template variables can be used in footers. Template Footers for details. PLEASE NOTE: this footer is NOT added to a message sent in 'text' only format, you must specify a text footer if you want a footer added to a text message (and a text footer can't contain html of course).
join_cookie_subject	false	true	If set then the cookie code is in the 'subject' of the message, this makes it easier for users to join a list. Join_cookie should also be set to true.
join_cookie	false	true	If set, when users join the list they will be asked to respond with a specific cookie (number) to prove they are real humans, this setting prevents people from subscribing other people or worse other lists to an existing list. Note: a cookie will not be sent if the subscriber is a moderator or if access_join for the list is set to moderator or password.

language_file (version 2.9d and above)	(none)	newproducts.dat	Used to specify a language file for a particular list. That language file is used to translate most of the phrases generated by DList for that list. Documentation on language translation is available here .
log_bounce (version 2.8h and above)	false	true	This is a debugging setting, that may be more generally useful. It causes DList to log all addresses that bounce and the reason for the bounce to the file, bounces.log, in the list home directory. The log is appended to for every bounce received when sending any messages to the list. See also, bounce_remove .
list	(none)	dnews-discussion	The name of the list, this cannot contain spaces and must be the first setting for each new list in lists.dat
max_size	150	500	Limits the maximum size of an item that can go through the mailing list in kbytes. NB: this setting applies to messages to the -request address as well as the posting address.
max_per_user (2.4g and above)	200 (changed from 50 in vers. 2.5d)	1000	Sets the max number of messages allowed to be posted to all lists on the server per user per hour. Note that the count is per user for posts to ALL lists, whereas the setting is per list. So the count is global but whether it applies to a list is list specific (the default is 200).
moderator	(none)	fred@netwinsite.com	A list of one or more moderator email addresses, a moderator often has extra access rights, like the ability to subscribe other people etc. Separate multiple entries with spaces or tabs (or commas in version 2.5d and above), emails are only sent to the first moderator in the list, but any moderator can send moderated messages.
no_processed_message (version 2.9d and above)	false	true	This setting is very powerful. If set to 'true' for a particular list, no command processed messages are sent by DList for that list. This means that users will only see list posts; they will get no response to any DList commands they send (i.e. who, lists etc.). This would only really be useful if you wanted to subscribe users to a list without them receiving notices.
no_welcome_message (version 2.9d and above)	false	true	By default, DList sends a welcome message to each user directly after that user successfully subscribes to a list. If set to 'true' for a particular list, then users subscribing to that list will not be sent a welcome message.
			If set, the reply-to header in each message will be pointed to the original poster, rather than the mailing list, this is recommended for large mailing lists.

reply_to_user	false	true (also in version 2.5f and above, user@domain)	In versions 2.5f and above in place of true you can specify an address as the reply address for ALL messages posted to the list. If given, posted messages will have any Reply-To: header turned into X-Reply-To:, and the address given is added to a new Reply-To: header.
status_interval	7	1	Period in days between automatic status reports being sent to the moderator.
skip_mailer_check	false	TRUE	If TRUE then DList will not ignore messages from users called, MAILER-DAEMON (all in capitals). These are normally bounced messages and so would not normally be wanted as posts to the list.
skip_postmaster_check	FALSE	TRUE	If TRUE then DList will not ignore messages from users called, POSTMASTER (all in capitals). These are normally bounced messages and so would not normally be wanted as posts to the list.
subject_prefix	(none)	Juggle:	This string will be added to the front of every subject line of messages from this list, this makes it easy for people to sort list messages out from other messages.
title	(none)	N.Z. Juggling	A title for the list, shown in headers and lists output.
invisible	false	true	Makes the list invisible to the 'lists' command for finding lists on your server.
mod_web	false	true	Add web links to accept/drop messages. Messages are stored so that the actual message with correct headers will go onto the list
mod_first	false	true	If user has never posted before send their post to the moderator for approval, on subsequent posts the user can post direct. (Useful to stop spam abuse of lists)
notify_joinleave	false	true	Send the list owner/moderator an email when users join/leave the list
auth_post	false	true	STRONGLY recommended for private lists and large mailing lists. Only allow posts from local authenticated users.
auth_who	false	true	STRONGLY recommended for ALL lists. Only allow who requests from local authenticated users.
auth_moderator	false	true	STRONGLY recommended for all lists where the moderator is a local user. Requires that the moderator uses smtp authentication to prove they are genuine.
to_user	false	true	If true then the To: address in each message sent will be the users email address instead of the list name

An example lists.dat file with entries for two lists, talk and juggling:

```

list talk
  archive true
  title The list for talkers.
  subject_prefix [list: talk]
  access_join Anyone
  access_post Moderator
  access_who Anyone
  access_get Anyone
  moderator talk.master@macro.com
  max_size 40
  footr c:\surgemail\dlist\talk\footer.txt
list juggle
  title The list for jugglers.
  subject_prefix [Juggle]
  access_join Anyone
  access_post Anyone
  access_who Anyone
  access_get Anyone
  moderator juggling.master@macro.com
  max_size 40
  footer c:\surgemail\dlist\juggle\footer.txt< /td>

```

Welcome Messages

DList comes with an example welcome message. It is stored in a file called join.tpl in a template format.

You can edit this template to the look that you require, and you can copy it to each list's directory (off the main DList directory) so that individual lists can have their own welcome message.

DList supports variables in templates as of 2.9g. All you need to do to use template variables is add the variables you want into the template. or information on how to use template variables, and a list of supported variable names, see [Template Variables](#).

The template files current supported are:

- join.tpl
- leave.tpl

These files can be in the main dlist directory, or in a specific dlist sub directory.

Adding users to a list

Usually users would add themselves to a list by sending a message to the list request address eg: listname-request@domain with the word subscribe in the message body.

DList will then add them to the users.lst file for that list. Users.lst for each list is stored in that lists directory (named after the list) off the DList directory (probably \surgemail\dlist\listname\users.lst or /usr/local/surgemail/dlist/listname/users.lst)

To add a number of users you have two options:

1. Add yourself as a moderator for the list and send the listname-request address a message with mutiple subscribe lines in the body.

So as a moderator you send the following email:

```

To: listname-request@domain
From: your_moderator_address

subscribe bob@domain1.com
subscribe judy@domain1.com

```

```
subscribe george@domain99.com
```

to join up the email addresses,
bob@domain1.com
judy@domain1.com
george@domain99.com

2. Directly edit the users.lst text file for the list and add the email addresses one per line.
So to add the same three users, you might edit the users.lst file to look like this:

```
u:tam@1.2.3.4 f:Tam Willacy p:0 t:0  
bob@domain1.com  
judy@domain1.com  
george@domain99.com
```

where the first line is an existing user on the list.

Don't worry about the format of lines for existing users. The next time DList has to write anything to the users.lst file it will add the email addresses that you have pasted/typed in correctly.

Adding Users' Real Names

To specify the users real name when you are subscribing them using either method, enter the users email address with the full name field as per an email client eg:

sending subscribe commands:

```
subscribe "bobby" bob@domain1.com  
subscribe "Judy Simpson" judy@domain1.com  
subscribe george@domain99.com "Georgie Porgy"
```

directly in users.lst:

```
u:tam@1.2.3.4 f:Tam Willacy p:0 t:0  
"bobby" bob@domain1.com  
"Judy Simpson" judy@domain1.com  
george@domain99.com "Georgie Porgy"
```

When the user subscribes themselves, the real name is taken from their email address (from the From header).

Moderated lists

There are several ways of posting into a moderated list depending on your settings:

access_post moderator

The simplest setting is: **access_post moderator** this is totally insecure. When a post comes in for the list the message is forwarded to the moderator, who then must post it again to the list. The 'from' address of the poster must match the moderator for the post to be accepted. Forging messages is so easy almost anyone can do this and post to your list directly!

access_post password

This setting relies on a password being set and then sent with every posted message, the password is stripped off the posted message, but this is very likely to fail with HTML messages so is also an unwise choice.

access_post moderator

auth_post true (or) auth_moderator true

This is the best setting, but it requires that moderated posts come from local users who are using smtp authentication, and also match the 'moderator' setting for the list.

Lastly for best security you can add "ALLOW_IP_MOD x.x.x.x" and list the ip address of the machine used to send the moderated messages.

Archives and Files

This is still being written :-)

DList lists can be set to save an archive of all messages by setting the list specific setting in lists.dat, [archive](#) true

If this is set then DList will create the archive messages in a subdirectory called 'archive' below the lists directory eg:
c:\surgeemail\dlist\listname\archive\1.msg
c:\surgeemail\dlist\listname\archive\2.msg
etc.

Then if the user sends an email to the listname-request address with the command, dir, in the message body DList will send back a message telling the user how many archived messages there are.

If the user wants one of the messages then they can send the 'get' command to fetch the archived message.

If you want to provide other files to the list members, then you create your own directory off the list's directory called, files, and put the files that you want to provide there eg:
c:\surgeemail\dlist\listname\files\picture.jpg

Then when the user does a 'dir' command they will also be shown a list of other files available.

For details on the list commands see, [DList Email Commands](#)

Template Footers

DList will treat footer files as templates. This means that you can use variable names in the footer file. These names will be replaced with the actual values as the message is sent. For information on how to use template variables and a list of supported variable names see [Template Variables](#).

Template Variables

Template variables are a way to customise joining messages and footers with information about a list or a list member. Variables are denoted in the following form:

%%variable_name%%

When DList comes across a variable in a template (say a footer or a join message) it replaces that variable with its value. For example, the following line contains two variables:

Welcome to the %%list_name%% list, %%h_user%%.

The variables are replaced with their actual values when the message is sent. As an example, the above line could become:

Welcome to the newproducts list, joe@bloggs.com.

Below is a list of all template variables supported by DList. Please note that some variables may have multiple variable names. This is to ensure backwards compatibility with older versions of DList.

Variable Name	Synonym	Replaced With	Example Value
list_member	h_user	The email address of the user to whom the message is sent.	joe@bloggs.com
list_request_address	list-request	The email address to which to send requests.	newproducts-request@bloggs.com
list_address		The email address to which to send list messages.	newproducts@bloggs.com
h_fullname		The users name, e.g. Joe Smith (if known in users.lst) You need to set body_template true to use template variables in messages you send	John Smith
list_name_only		The list name without the domain name	newproducts
list_title		The 'title' of the list	New Products

list_name	list list-name	The name of the list. Which is generally identical to the list_address (this variable exists for historical reasons, you probably want list_name_only instead)	newproducts@bloggs.com
-----------	-------------------	--	------------------------

Some obscure/internal information

Digests

DList supports daily digests which are a compendium of all messages sent to the list within the past 24 hours.

The format of the users.lst file is very important. It is automatically generated, but for bulk additions you may need to modify it manually. If so, the format must be exact. Note: tabs are required - spaces will not be read as separating characters since they can be part of a full name. Also note this format is subject to change, the information here is provided as a guide, and it's fairly likely to remain accurate, but be warned :-)

u:(data)<tab>f:(data)<tab>p:(data)<tab>t:(data)
u: is the email address of the subscriber

f: is the full name of the subscriber
p: is an automatically generated and used for password verification subscriptions

t: is the type of subscription it is a bitmask.
1 is digest delivery
2 is disabled (no delivery)
4 is holiday (no delivery but user can re-enable)

Examples:
u:chris@xnetwin.xco.nz f:Chris P p:5112 t:1
u:robert@xtellurian.xcom f:Robert Boyle p: t:0

Chris P is a digest subscriber he will receive one compiled post at midnight at chris@xnetwin.xco.nz.

Robert Boyle is an immediate subscriber he will receive posts immediately as they are sent to the list at robert@xtellurian.xcom

Note:

When creating automatic scripts to subscribe an email address, the body of digest subscriptions messages to the listname-request address must be formatted as follows:

subscribe
digest true

This will subscribe the user and then set the digest mode for that subscription.

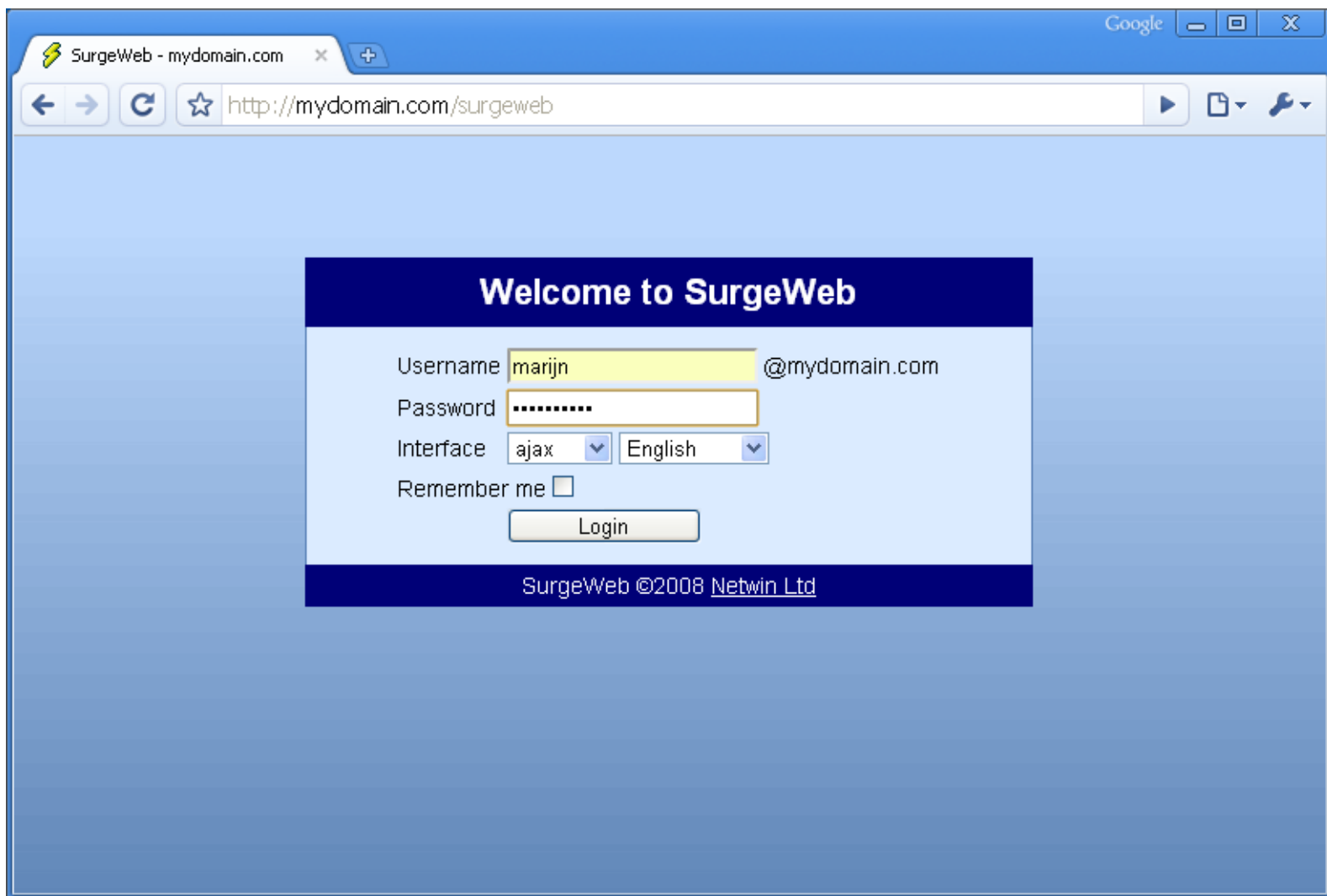
digest_textonly true - This option will strip all non-text MIME parts and only use the text parts to make the digest.

list_textonly true - This option will strip all non-text MIME parts and forward only the remaining text message to the list.

text_digest true - This makes the digest format simple non mime which many mail clients cope with better

The following alias addresses exist these allow you to join, leave and join for 'digests' by simply sending email to an address, with no 'body' content.

- listname-join@domain.name = Join the mailing list.
- listname-digest@domain.name = Join the mailing list, and set to receive 'digests'
- listname-leave@domain.name = Leave the mailing list.



Google

SurgeWeb - mydomain.com

← → ↺ ☆

http://mydomain.com/surgeweb

▶ 📄 ⚙

marijn@mydomain.com | [Options](#) | [Help](#) | [Logout](#)

SurgeWeb

Mail

Compose

▼ **Inbox (9)** [Refresh](#)
Inbox subfolder

Drafts

Sent

Trash [Purge](#)

▼ **More Folders** [Manage](#)
Deleted Items
Folder 1
Folder 2
Folder 3
Junk E-mail
Sent Items

Mail

Calendar

Addressbook

Photos

Blogs

Reply Forward Delete Move to... More Actions

Quick search...

1 - 14 of 14

Inbox Select

<input type="checkbox"/>	Richey	[SurgeMail List] Server Backups	5 KB	4:24 PM	▲
<input checked="" type="checkbox"/>	test1@orion	mid 0 testmail message	1 KB	3:14 PM	
<input type="checkbox"/>	SurgeMail Support	[SurgeMail List] Re: Problem with new mail server	3 KB	1:07 PM	
<input type="checkbox"/>	Dave Collar	RE: [SurgeMail List] Re: Mirroring	8 KB	7:57 AM	
<input type="checkbox"/>	SurgeMail Support	Re: [SurgeMail List] Re: Mirroring	2 KB	7:10 AM	
<input type="checkbox"/>	SurgeMail Support	Re: [SurgeMail List] Re: Mirroring	2 KB	6:38 AM	
<input type="checkbox"/>	Surgemail Support (Mar	Re: [SurgeMail List] webmail	3 KB	4 Nov	
<input type="checkbox"/>	Surgemail Support (Marijn)	Re: [SurgeMail List] How to use webmail filters to reply w	4 KB	4 Nov	
<input type="checkbox"/>	Robert Fisher	Re: [SurgeMail List] dufus customer..... how do i fix	6 KB	4 Nov	▼

From: test1@orion

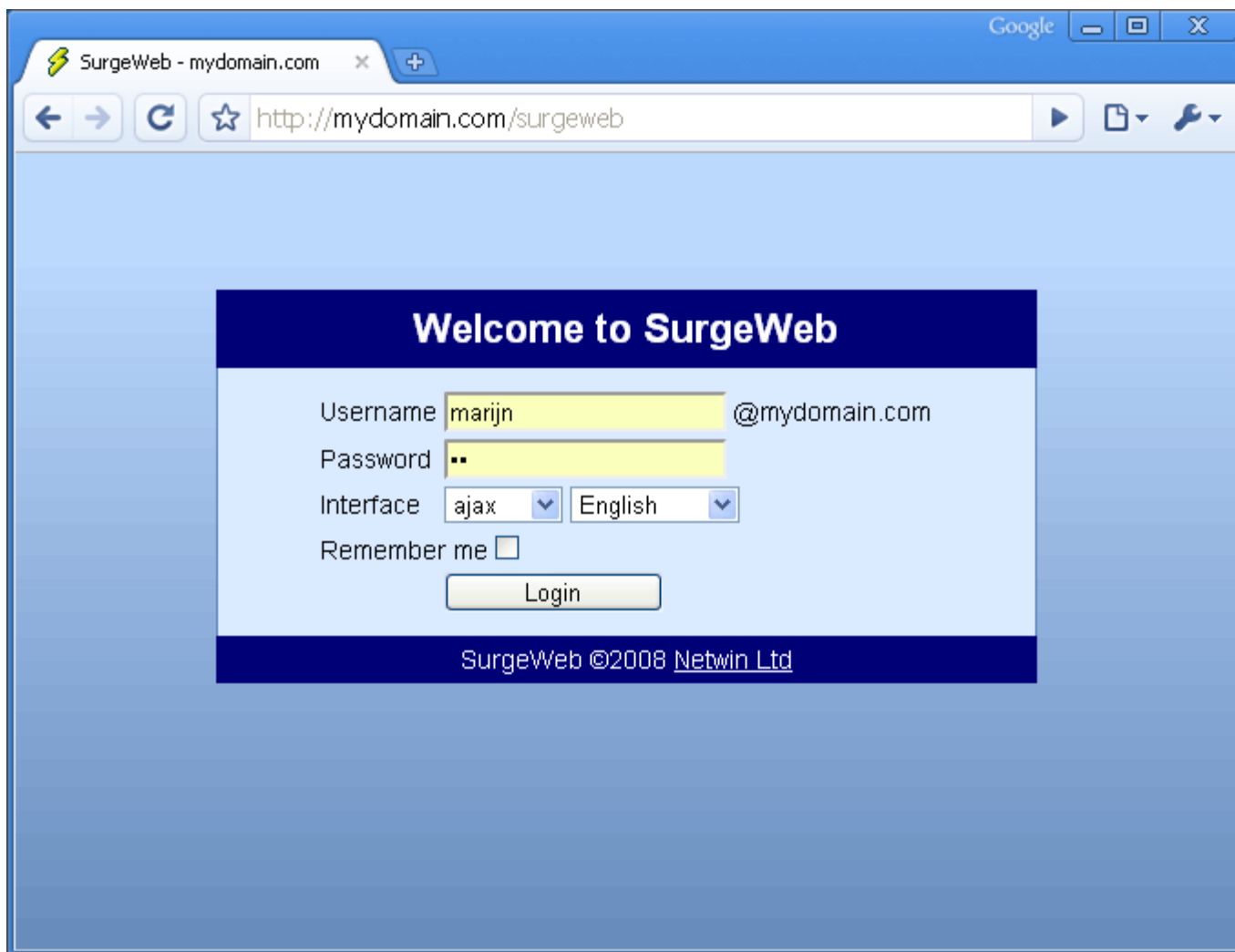
Subject: mid 0 testmail message

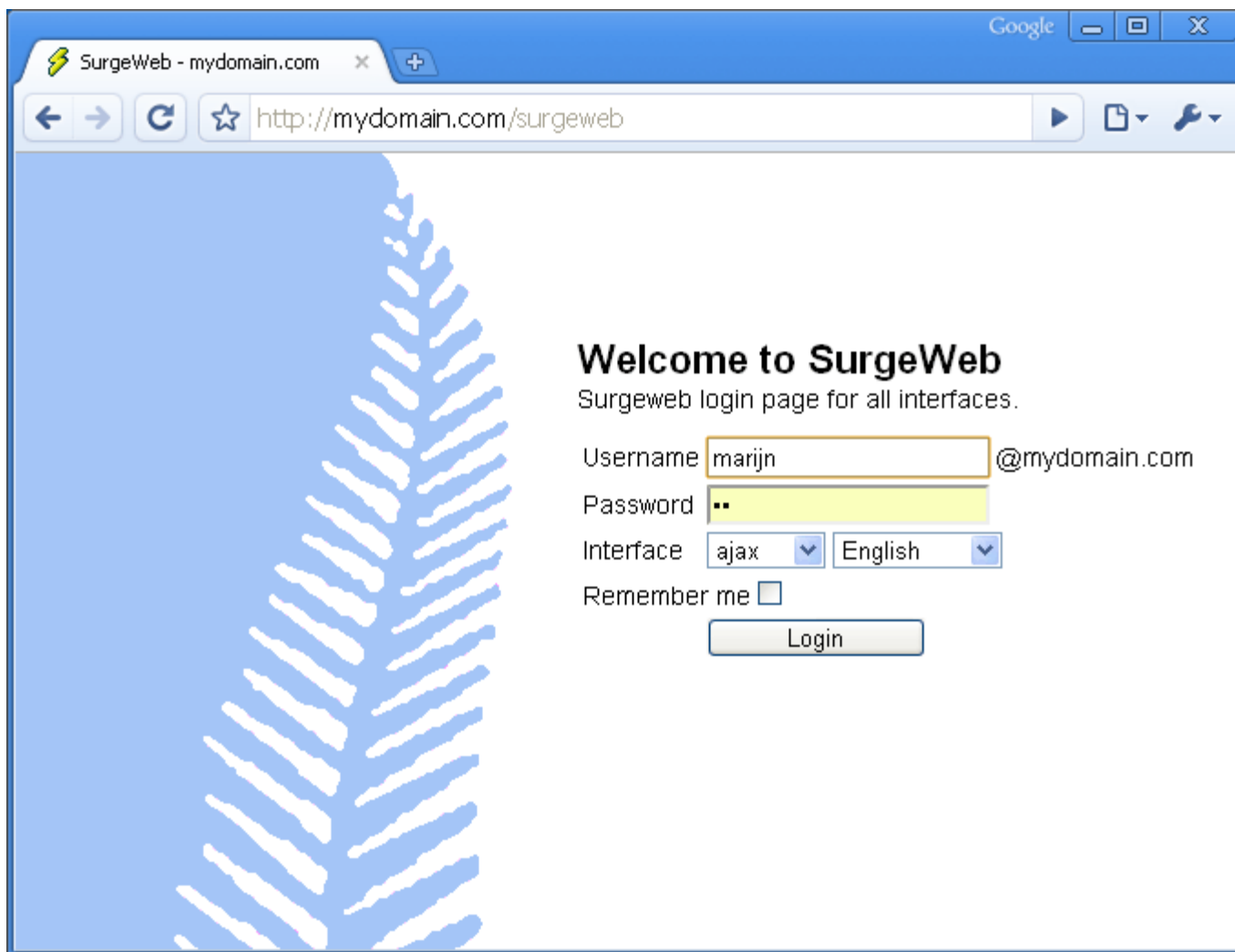
Date: 05/11/2008 3:14 PM

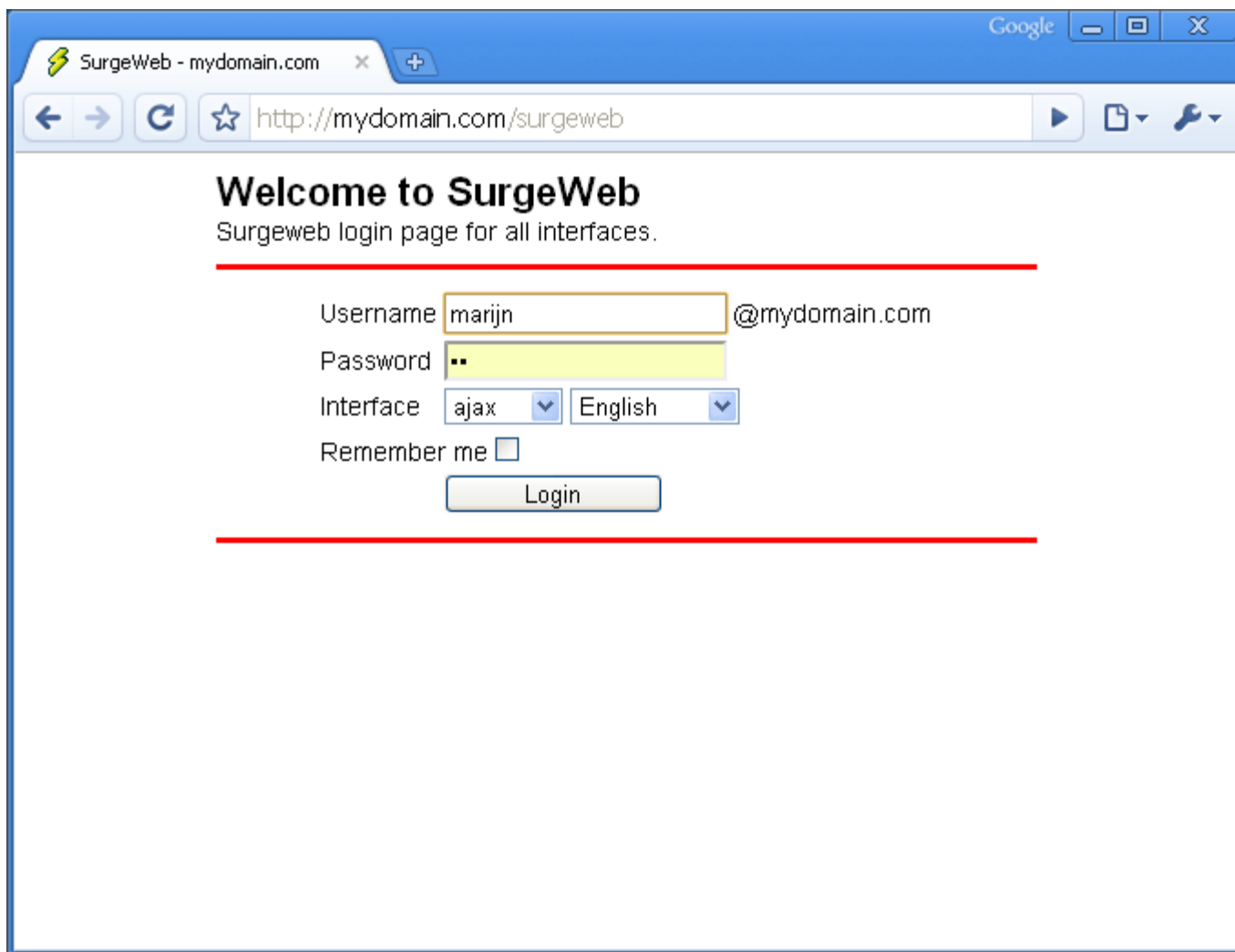
Show: [raw](#) [text](#)

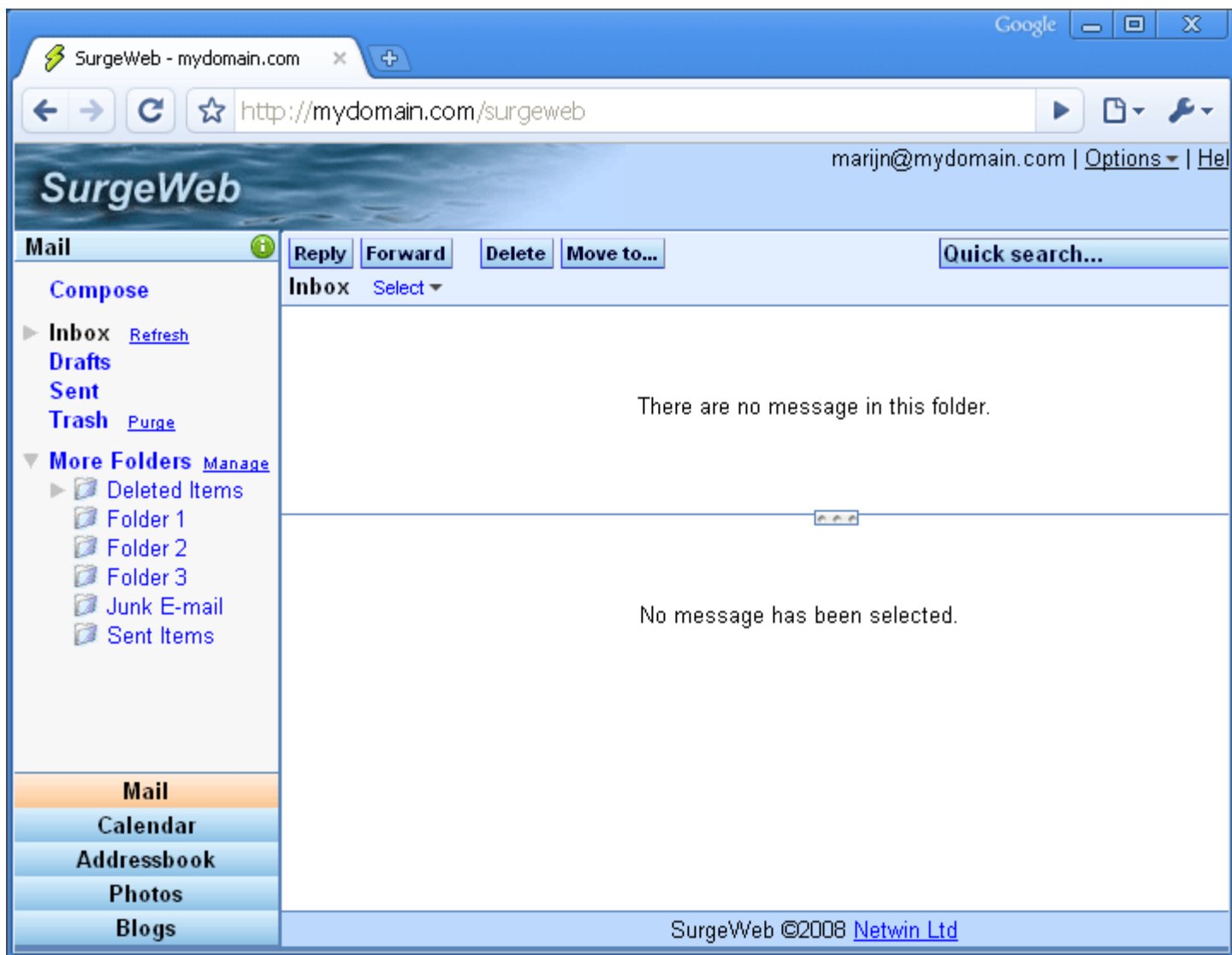
Body 0
1 aa
2 aa
3 aa
4 aa
NOCRC_ENDMID 0 lines 0 bites 0 ctot 0

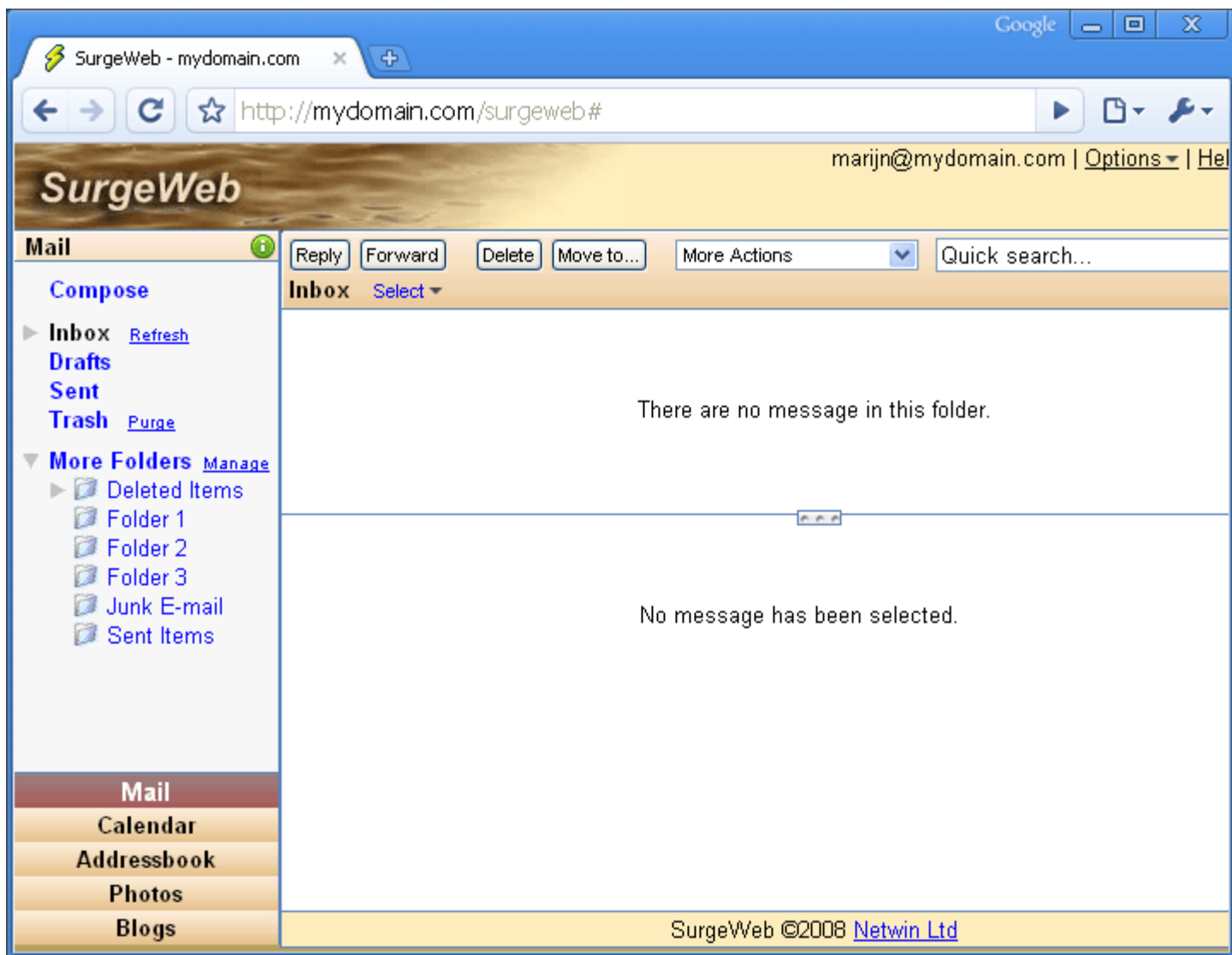
SurgeWeb ©2008 [Netwin Ltd](#)

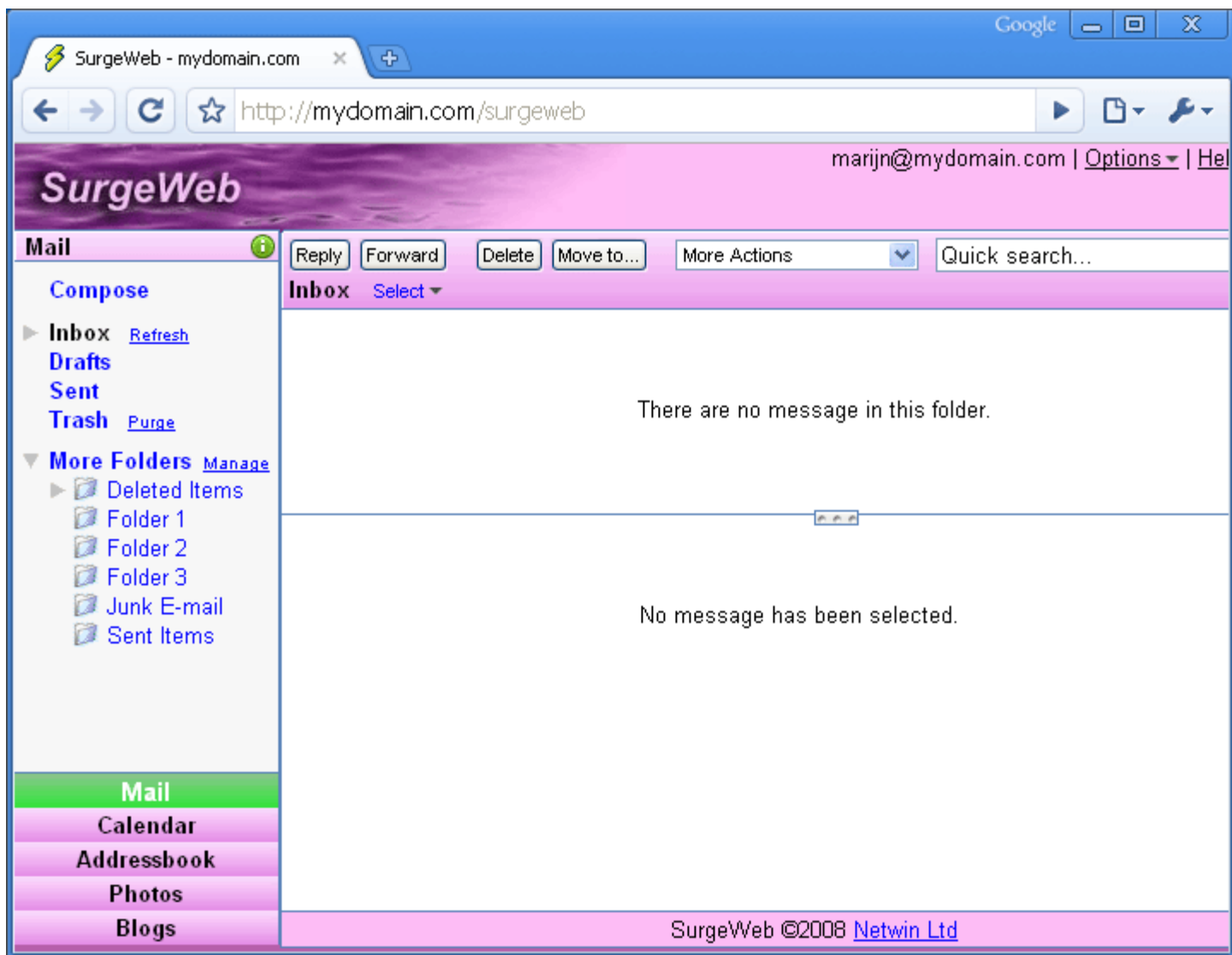


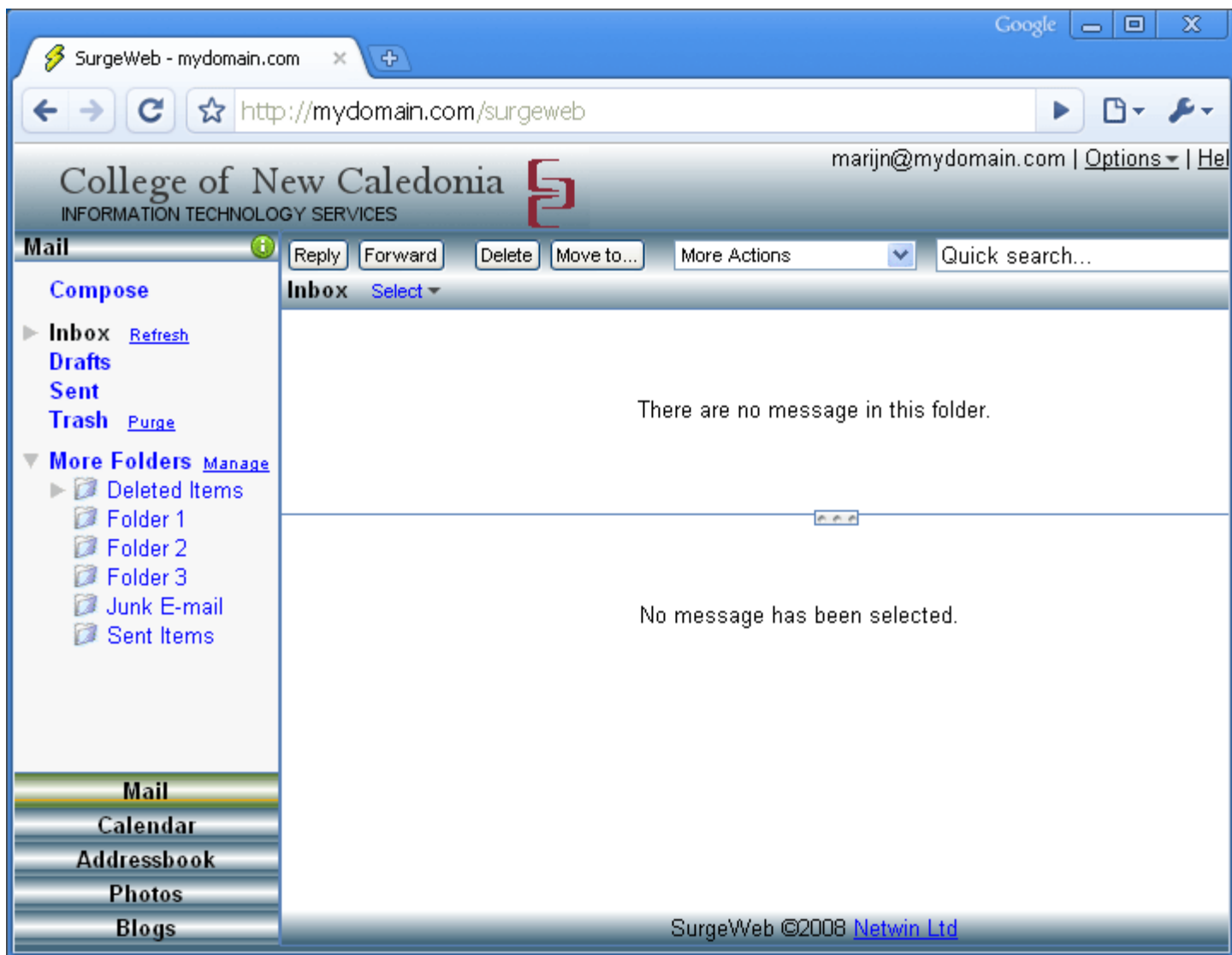


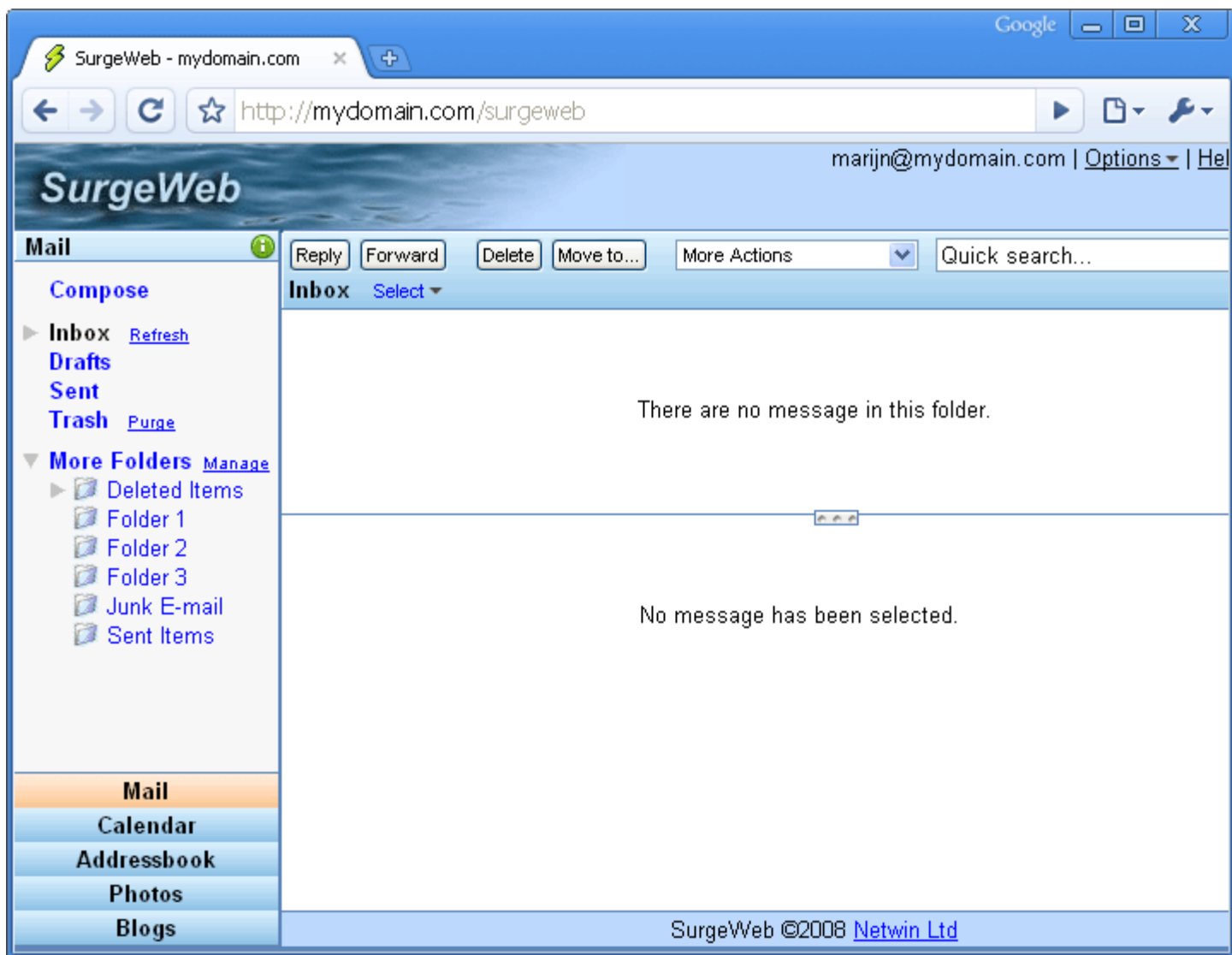


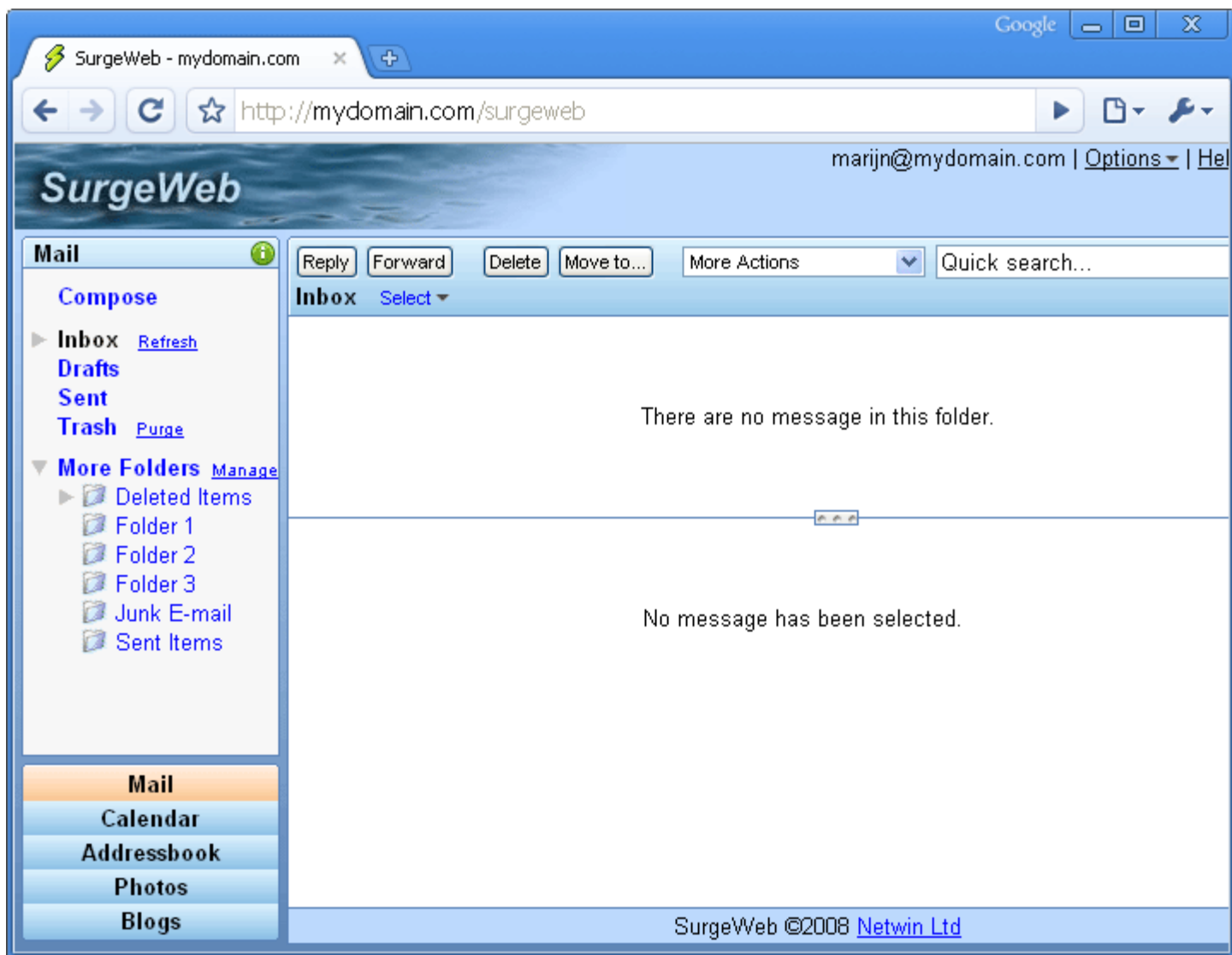


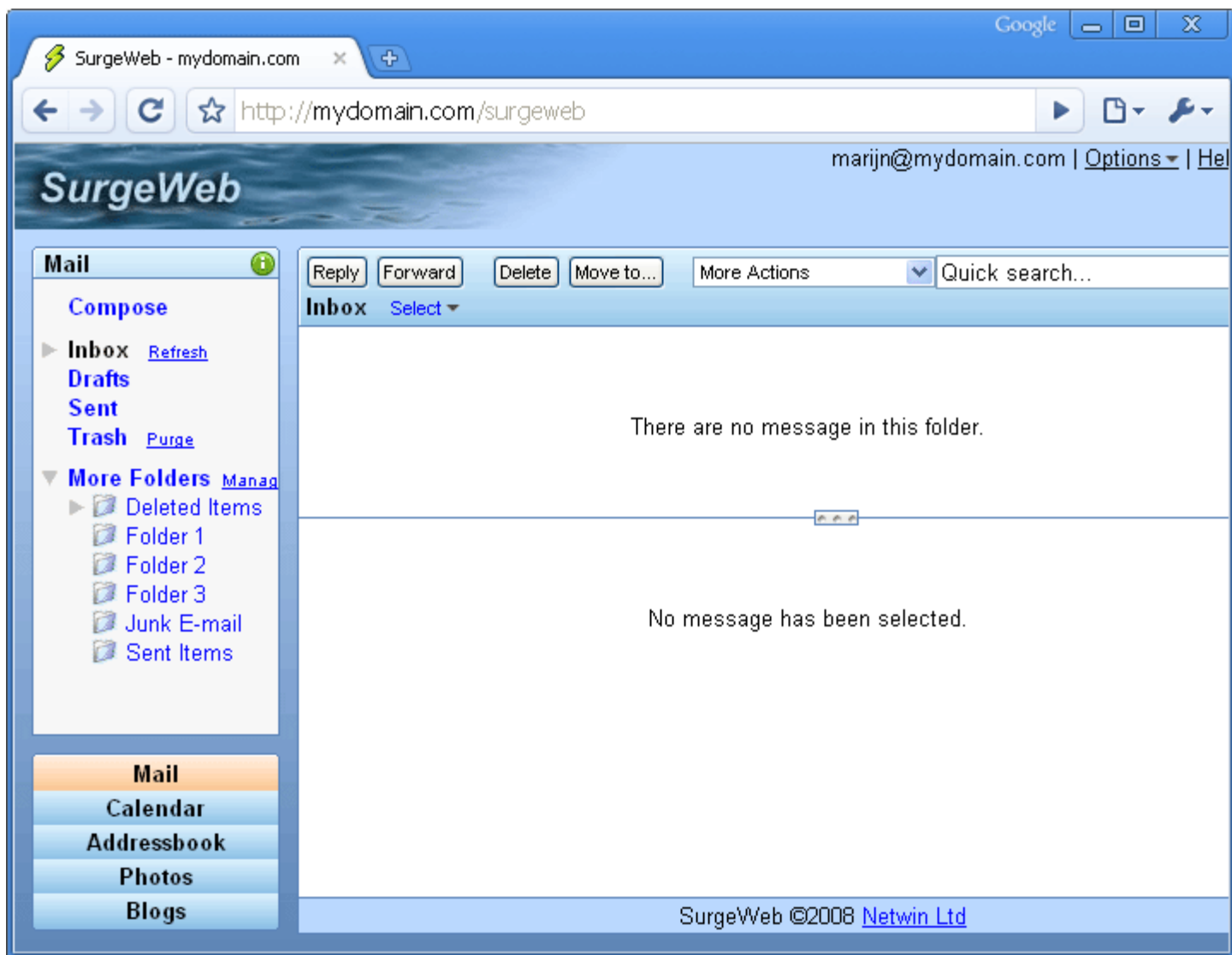


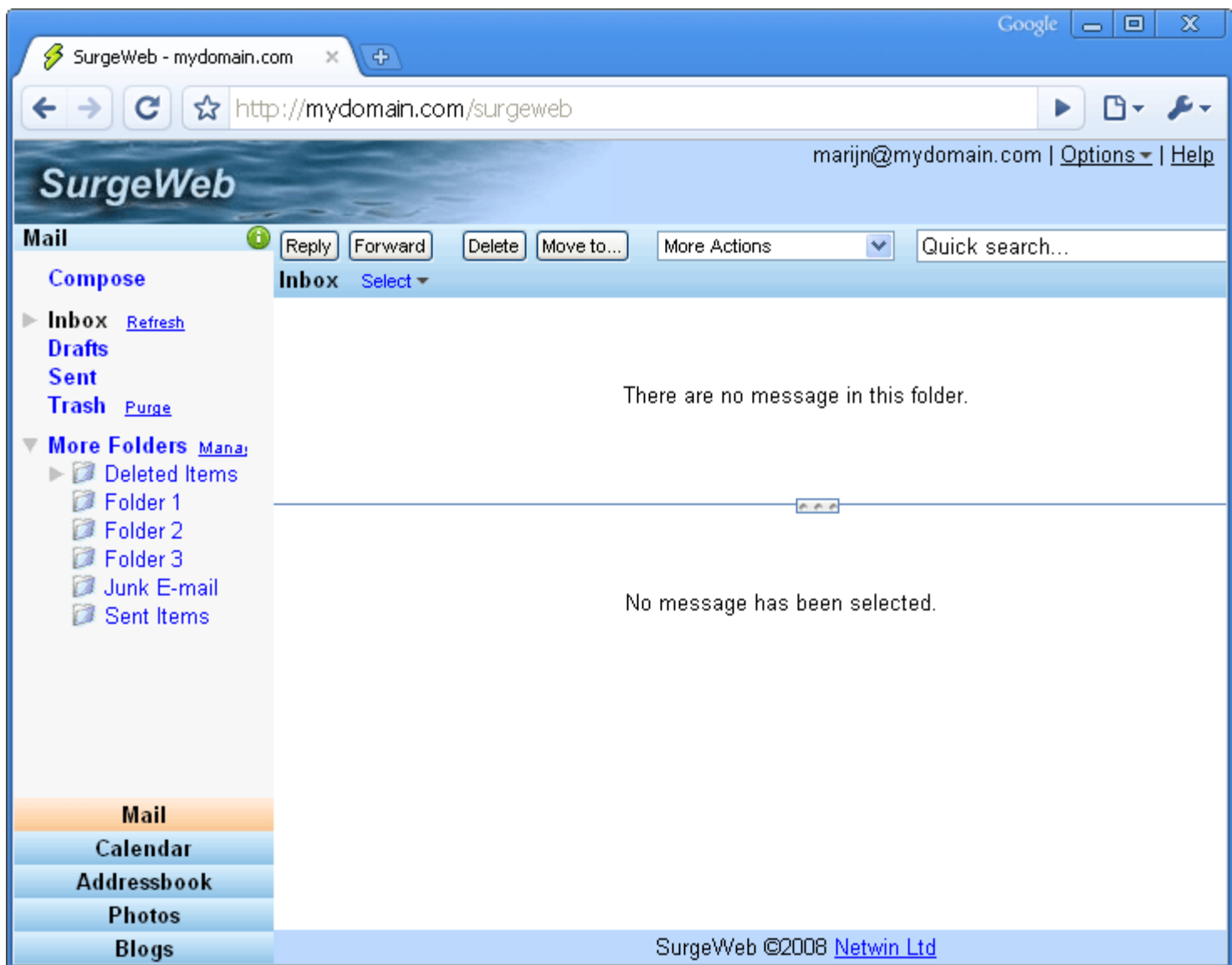


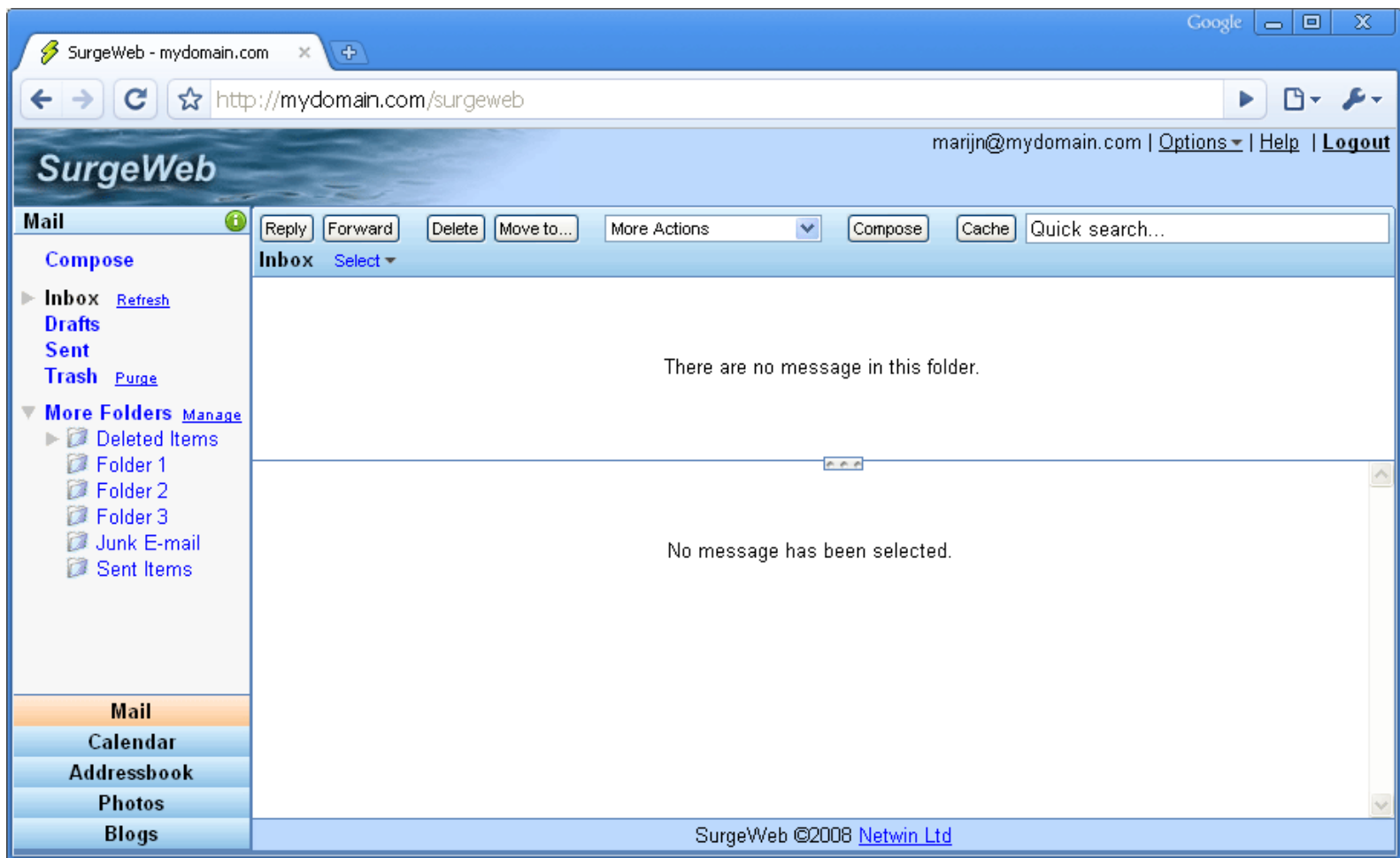


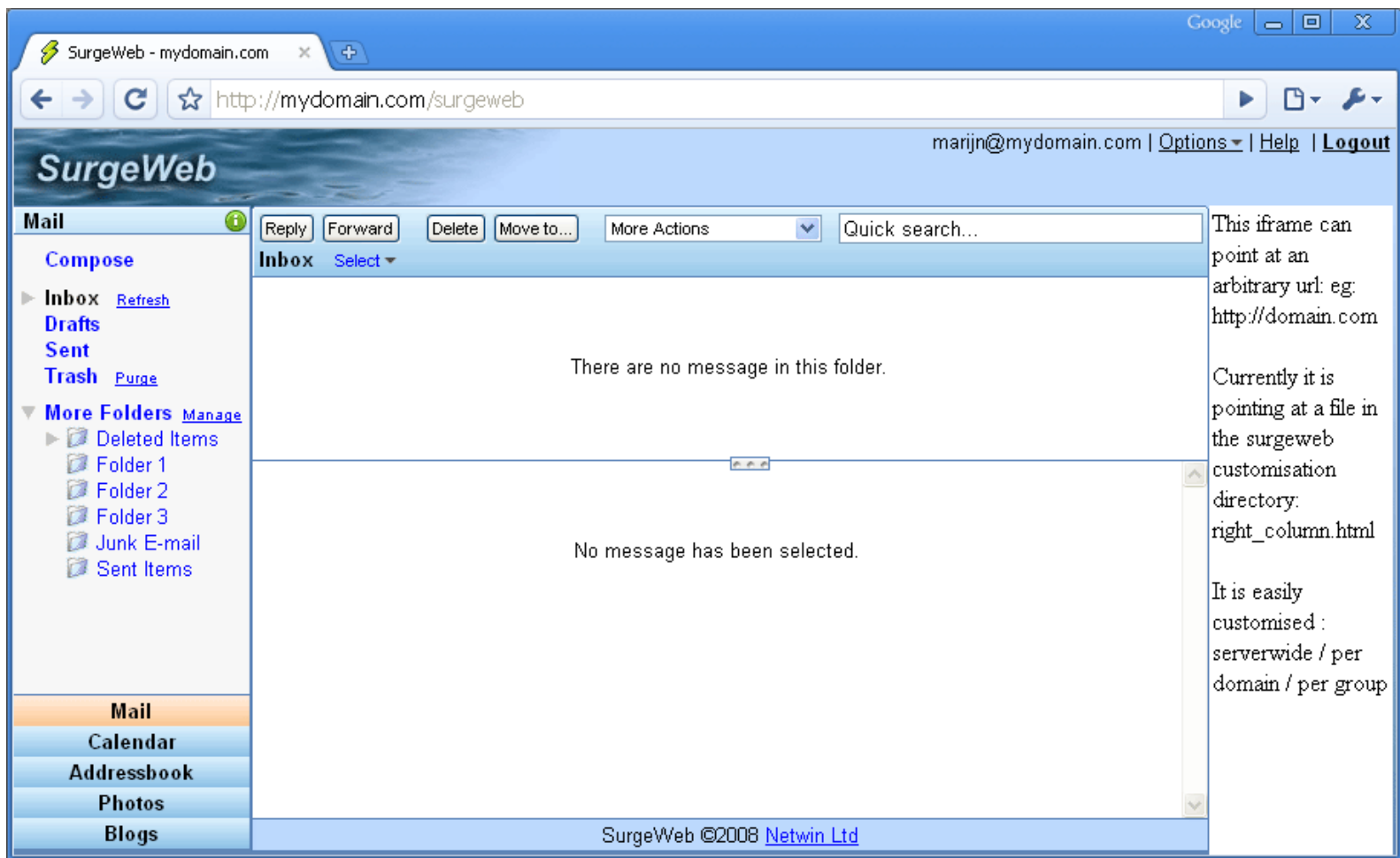








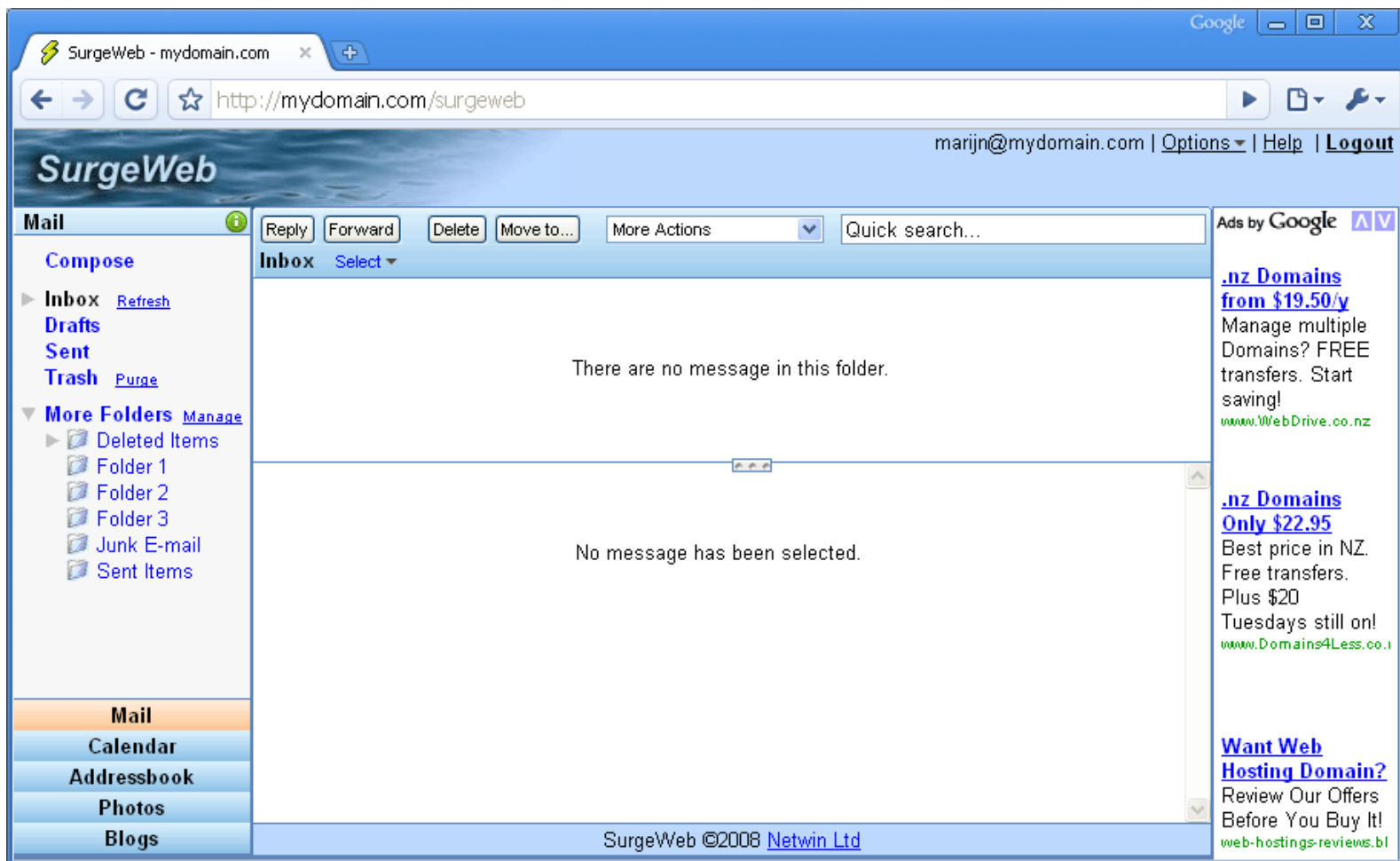


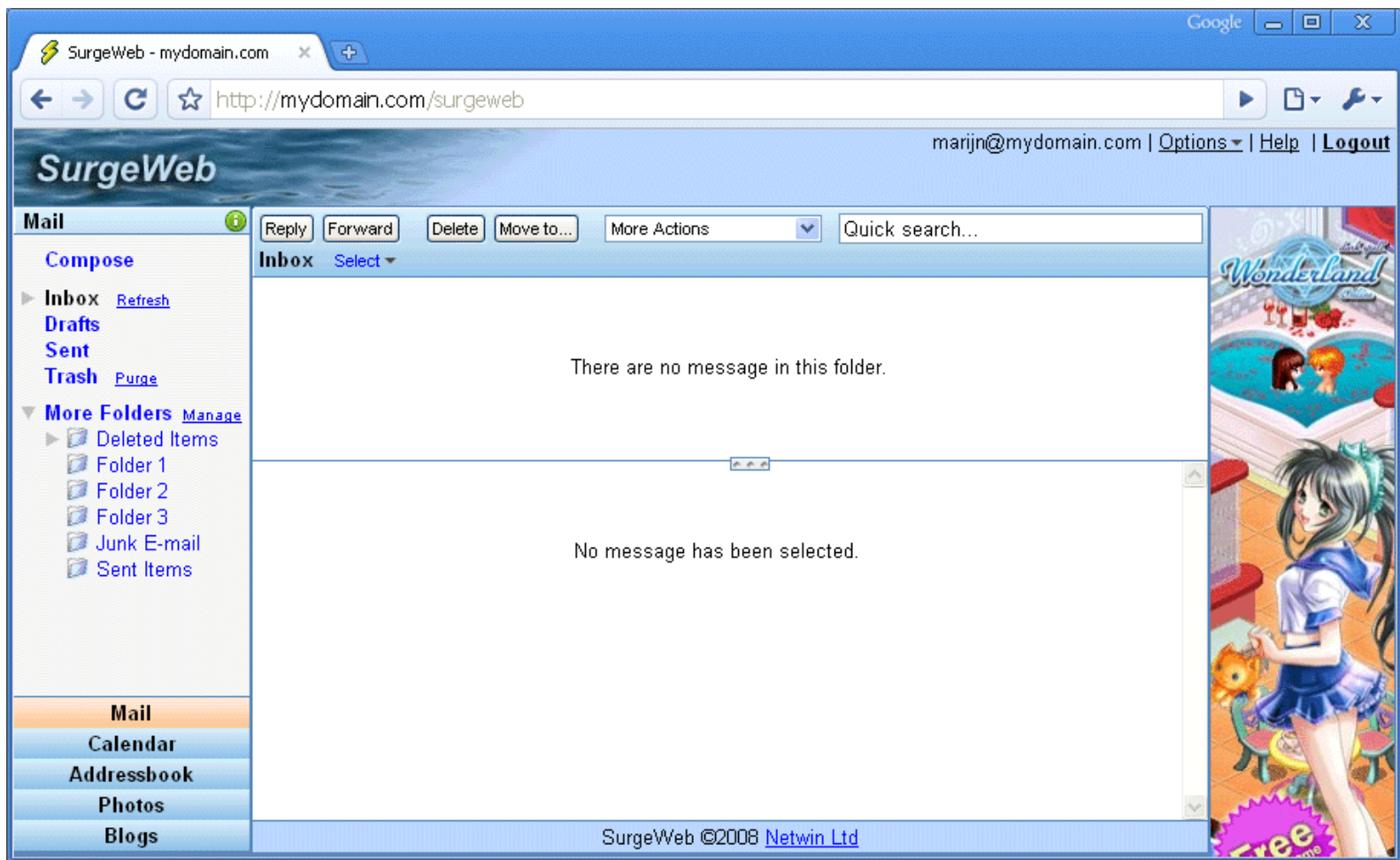


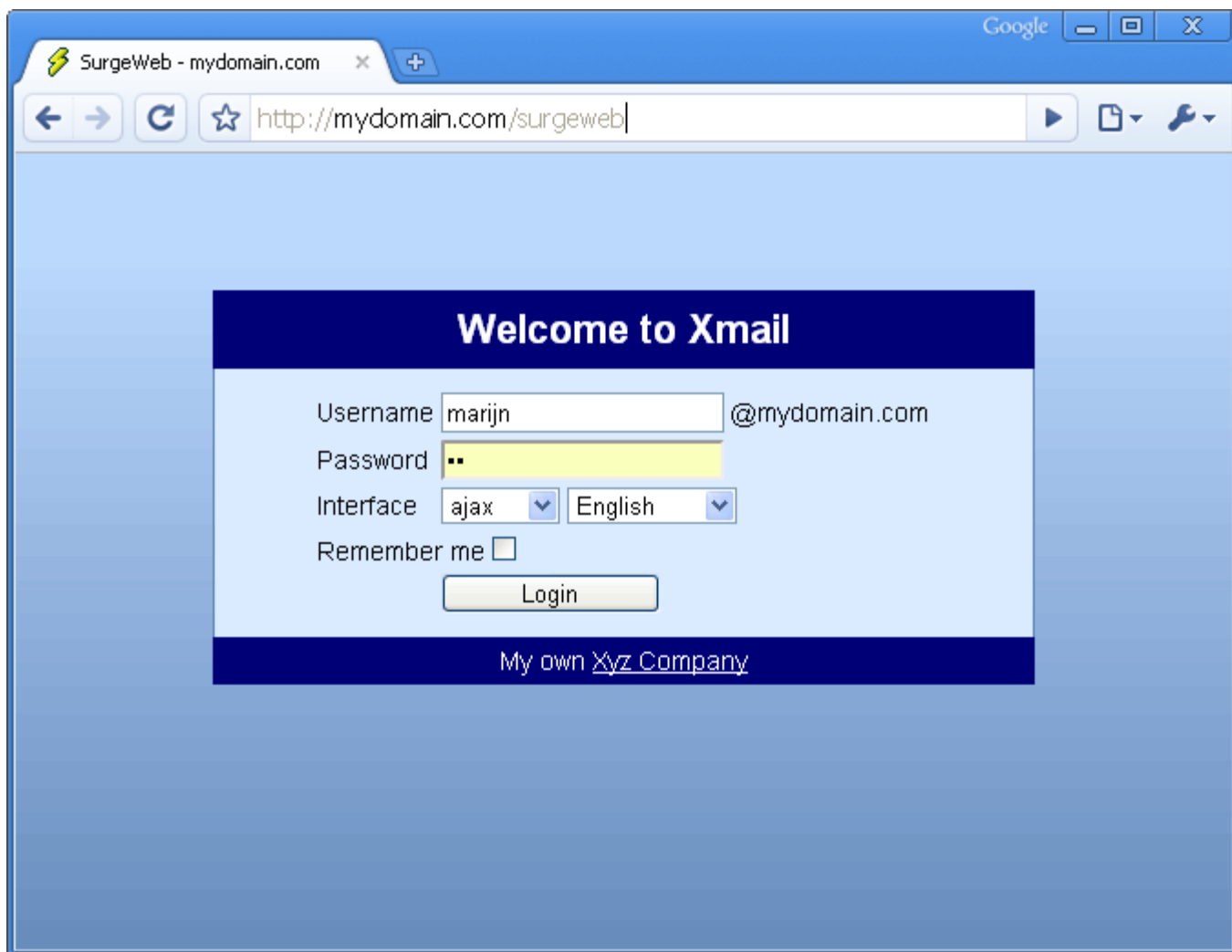
This iframe can point at an arbitrary url: eg: <http://domain.com>

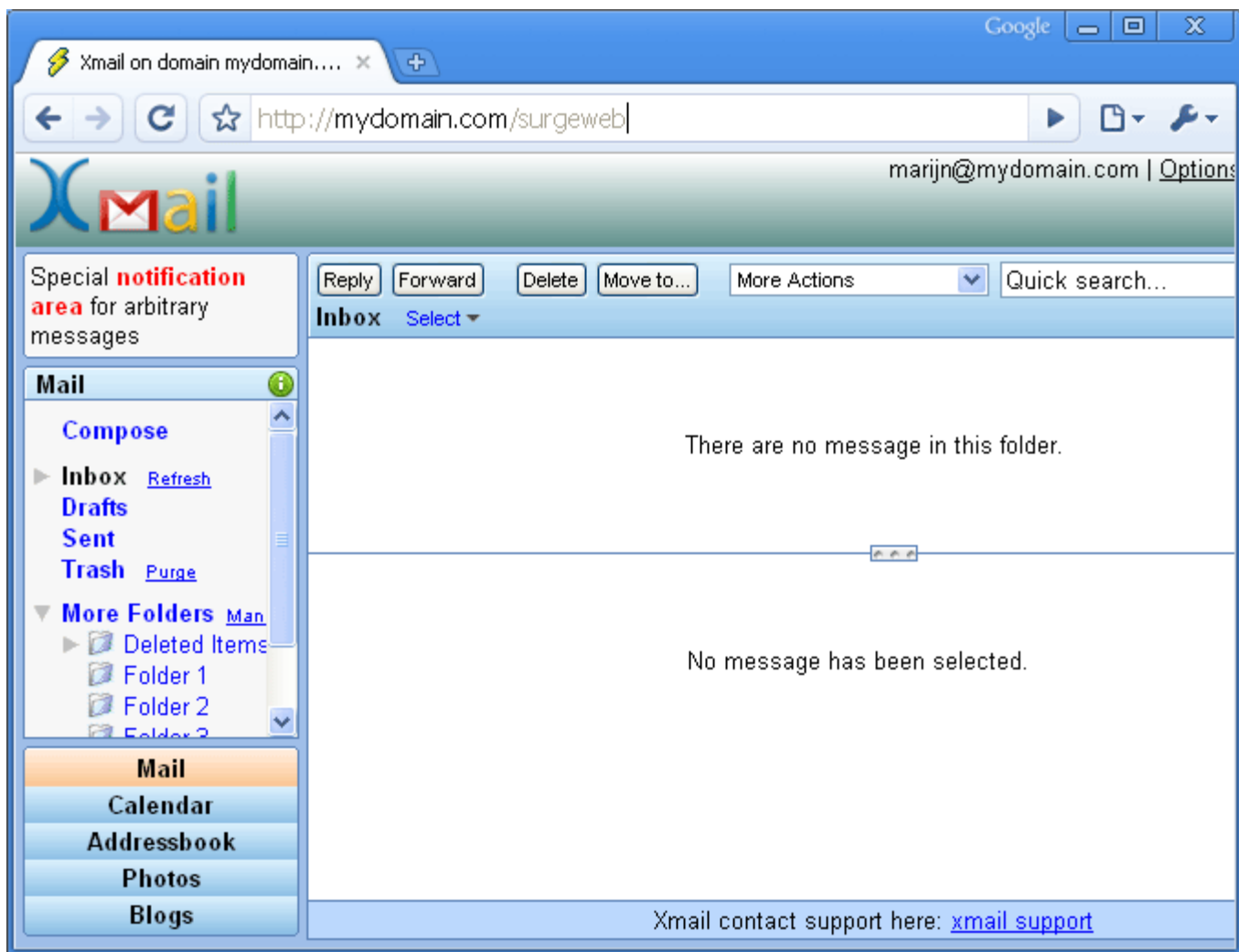
Currently it is pointing at a file in the surgeweb customisation directory: [right_column.html](#)

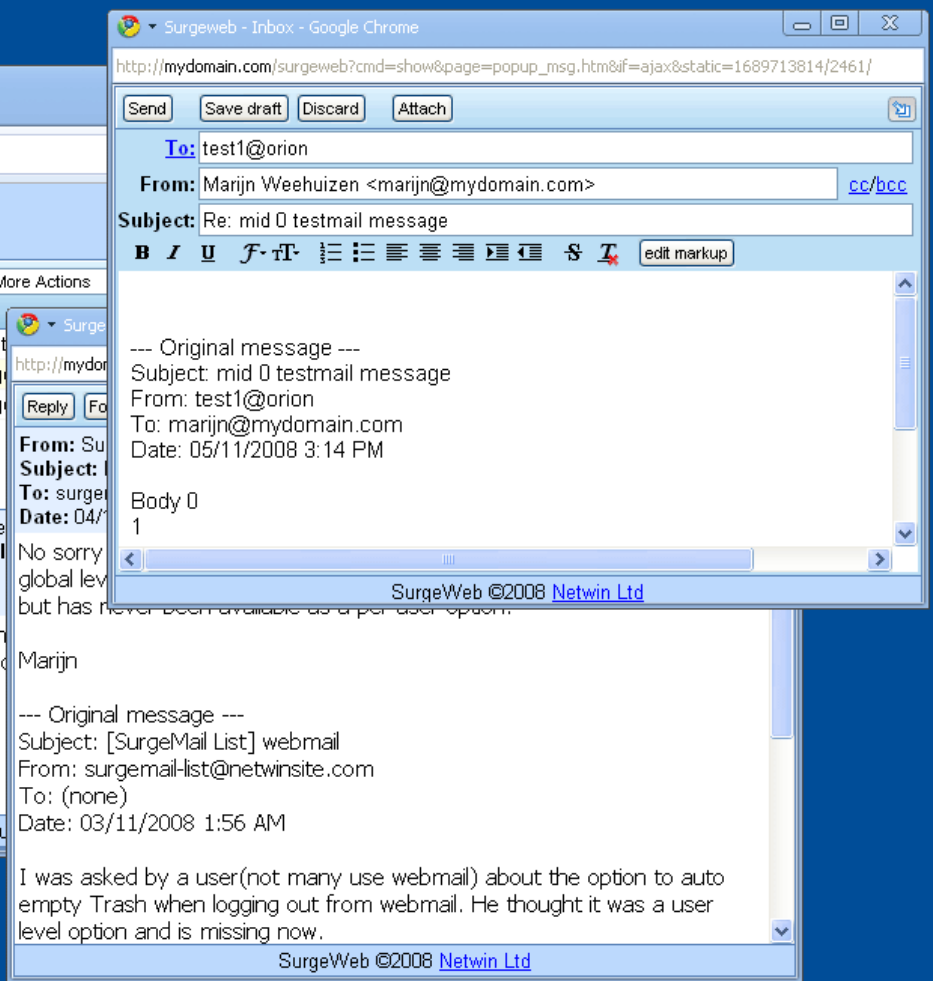
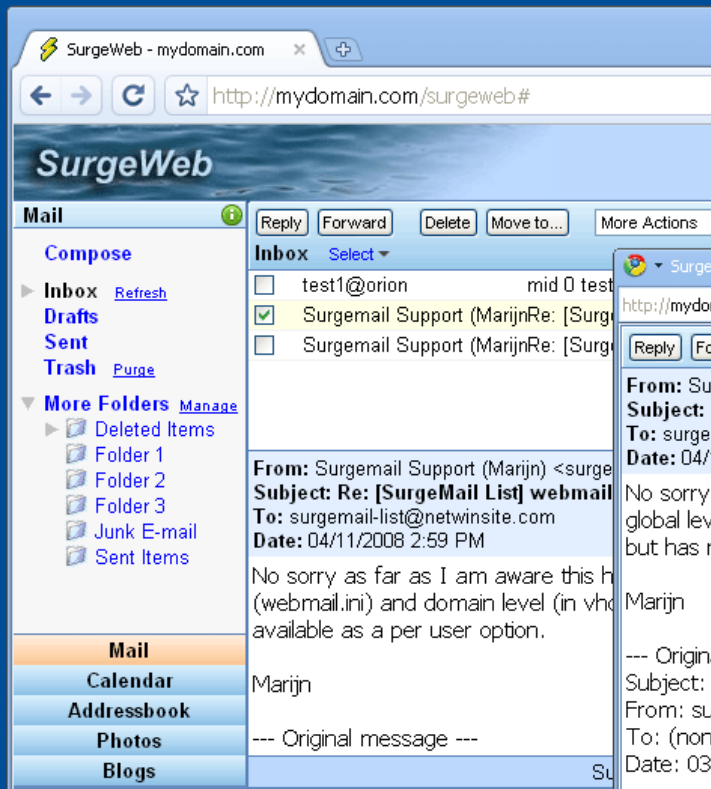
It is easily customised :
serverwide / per domain / per group



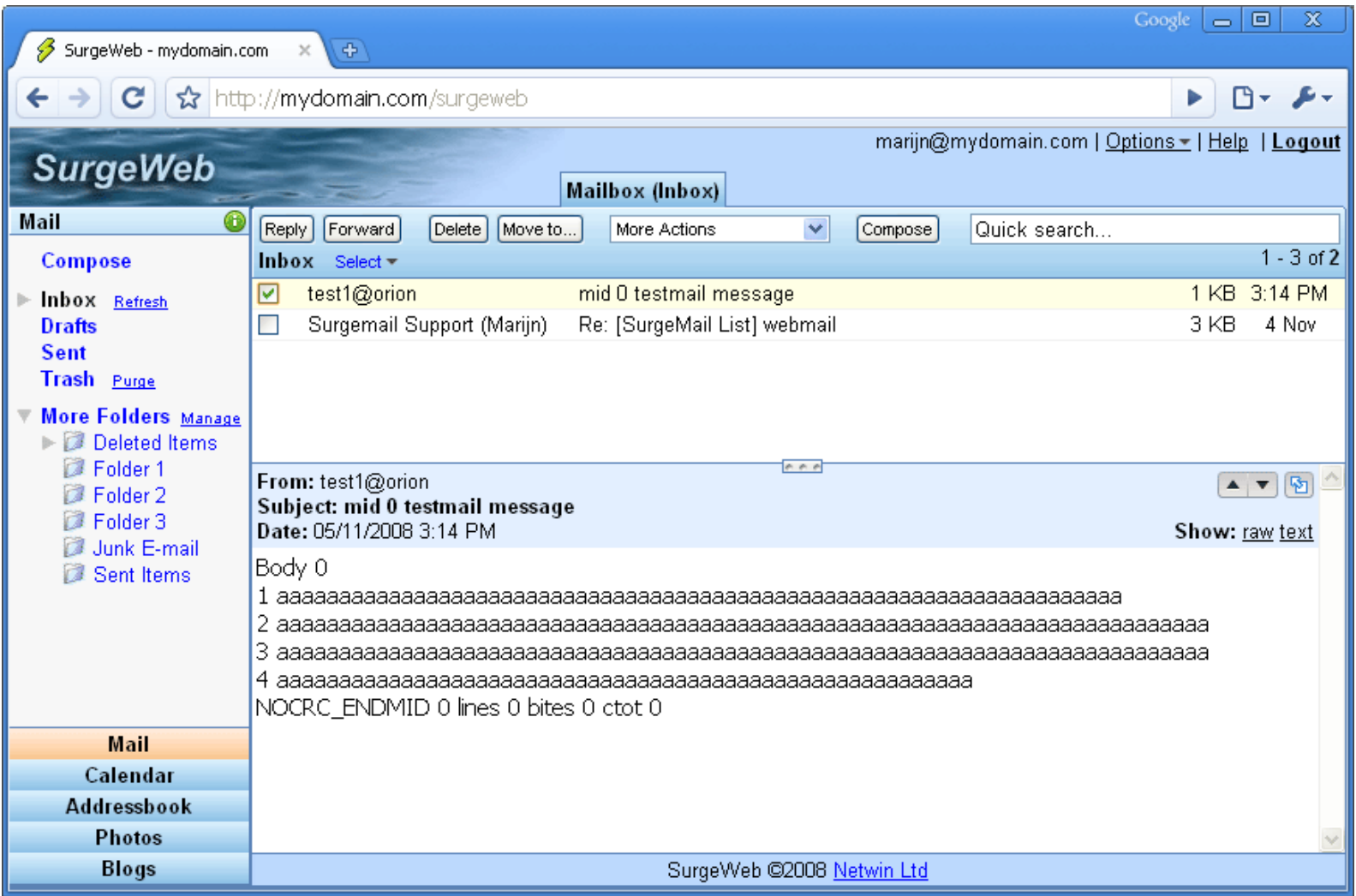


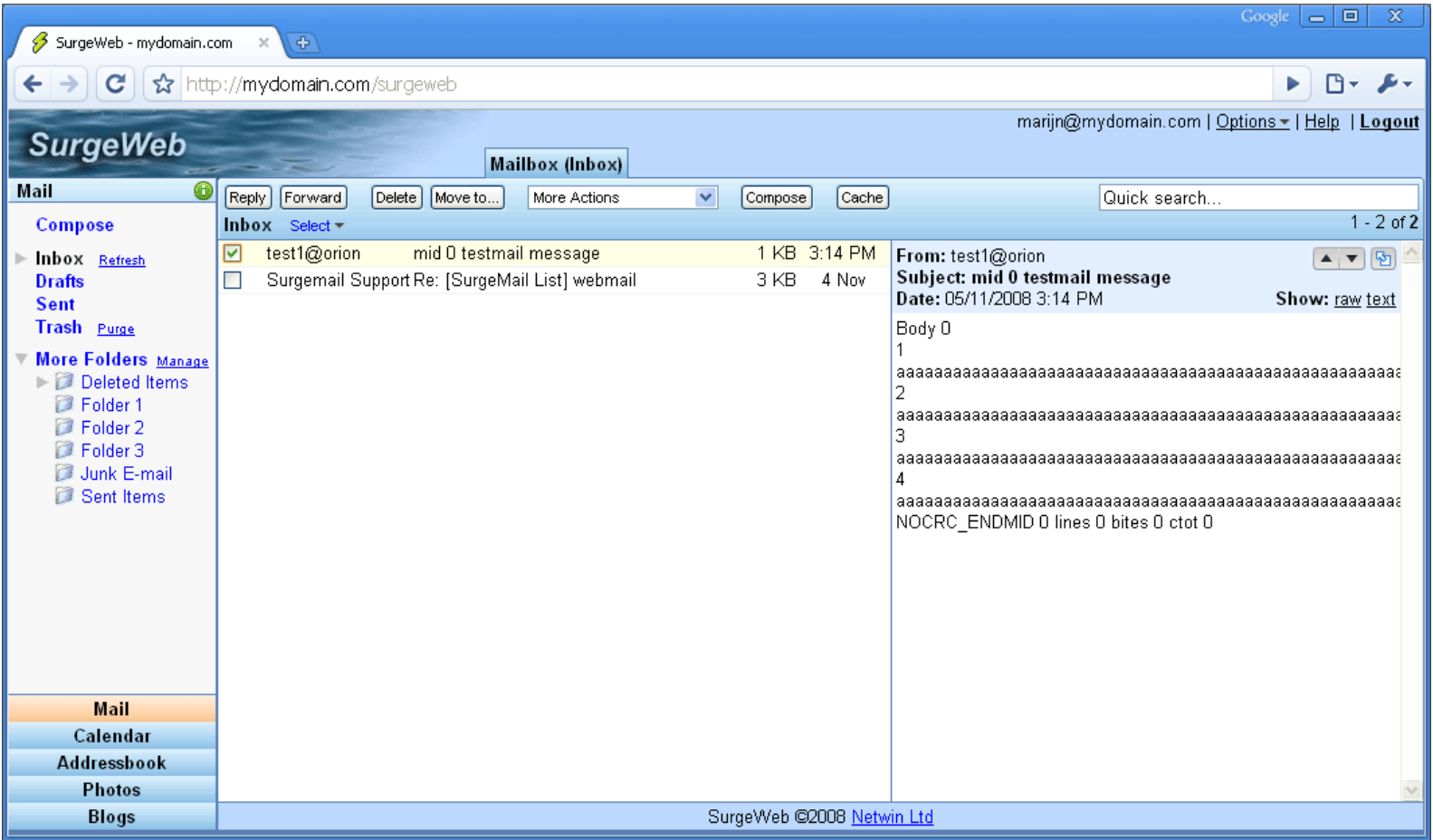


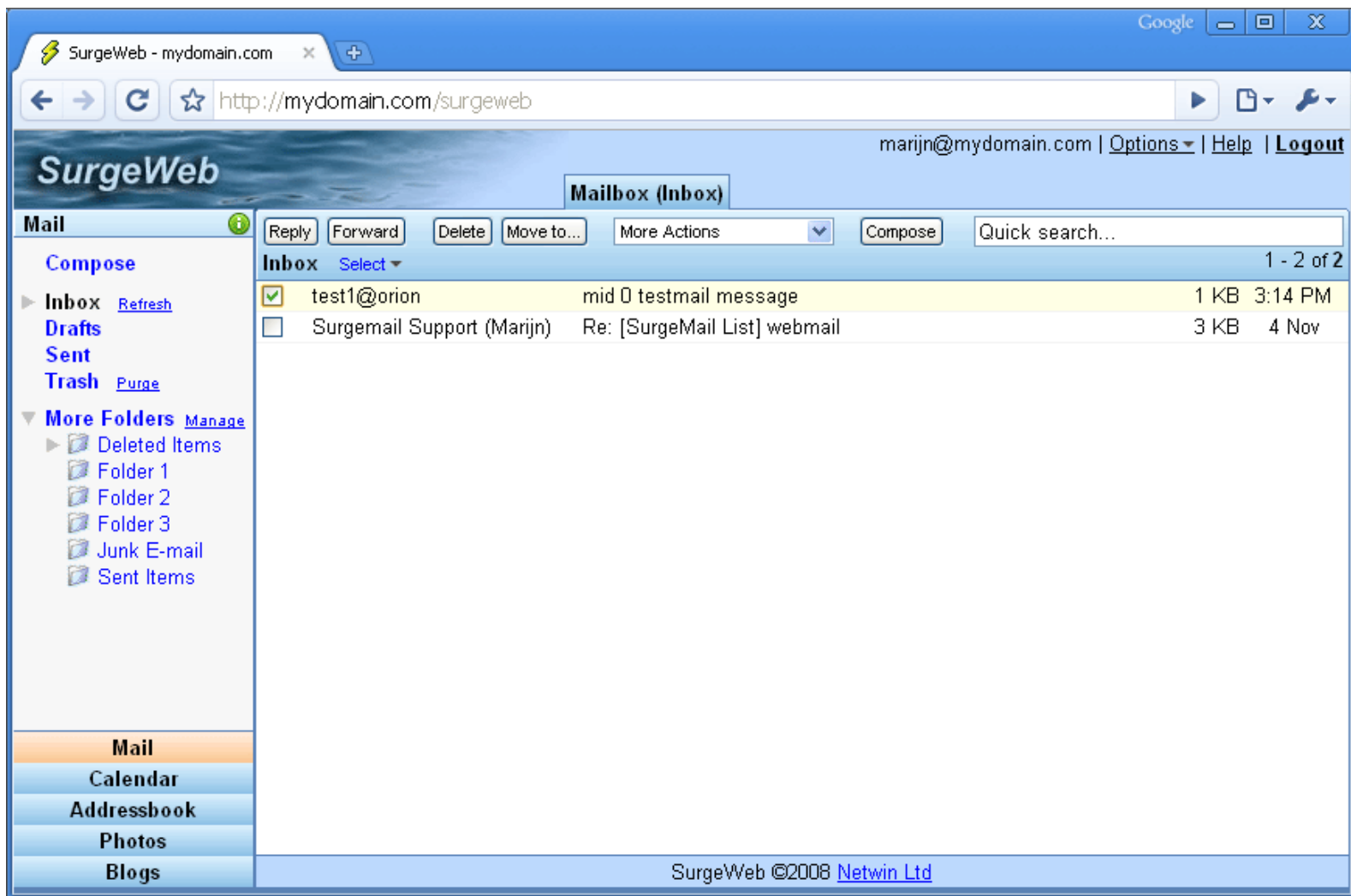












Google

SurgeWeb - mydomain.com

←

→

↺

☆

http://mydomain.com/surgeweb

▶

📄

🔧

marijn@mydomain.com | [Options](#) | [Help](#) | [Logout](#)

SurgeWeb

Mailbox (Inbox)

Mail

Compose

▶ Inbox [Refresh](#)

Drafts

Sent

Trash [Purge](#)

▼ More Folders [Manage](#)

▶ Deleted Items


Folder 1

Folder 2

Folder 3

Link Email

WikiMapia!! [link](#)



[Google](#)

[sciences Pty Ltd - Terms of Use](#)

Mail

Calendar

Addressbook

Photos

Blogs

Reply

Forward

Delete

Move to...

More Actions

Compose

Quick search...

Inbox

Select

1 - 2 of 2

<input checked="" type="checkbox"/>	test1@orion	mid 0 testmail message	1 KB	3:14 PM
<input type="checkbox"/>	Surgemail Support (Marijn)	Re: [SurgeMail List] webmail	3 KB	4 Nov

From: test1@orion

Subject: mid 0 testmail message

Date: 05/11/2008 3:14 PM

Show: [raw](#) [text](#)

Body 0

1 aa

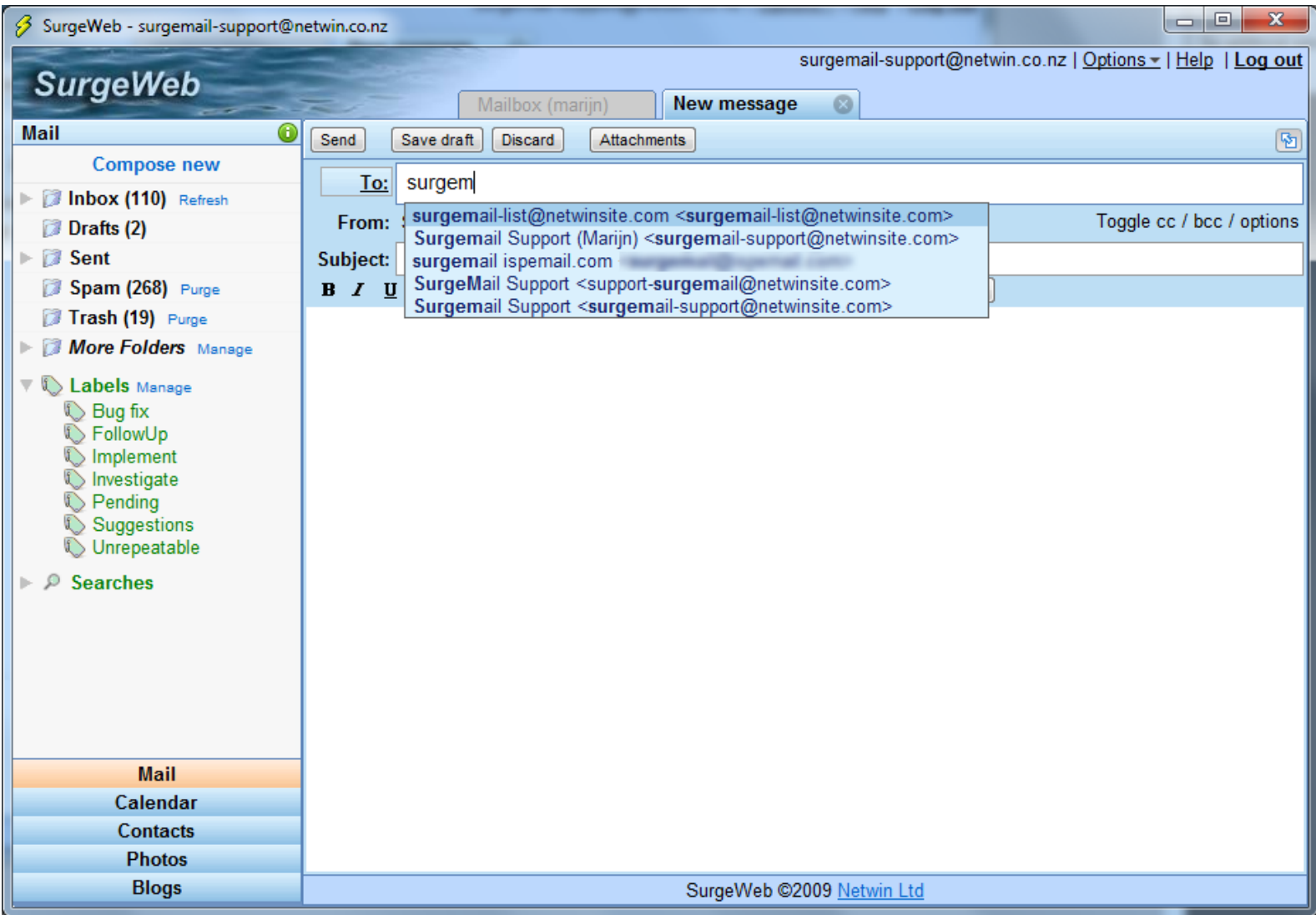
2 aa

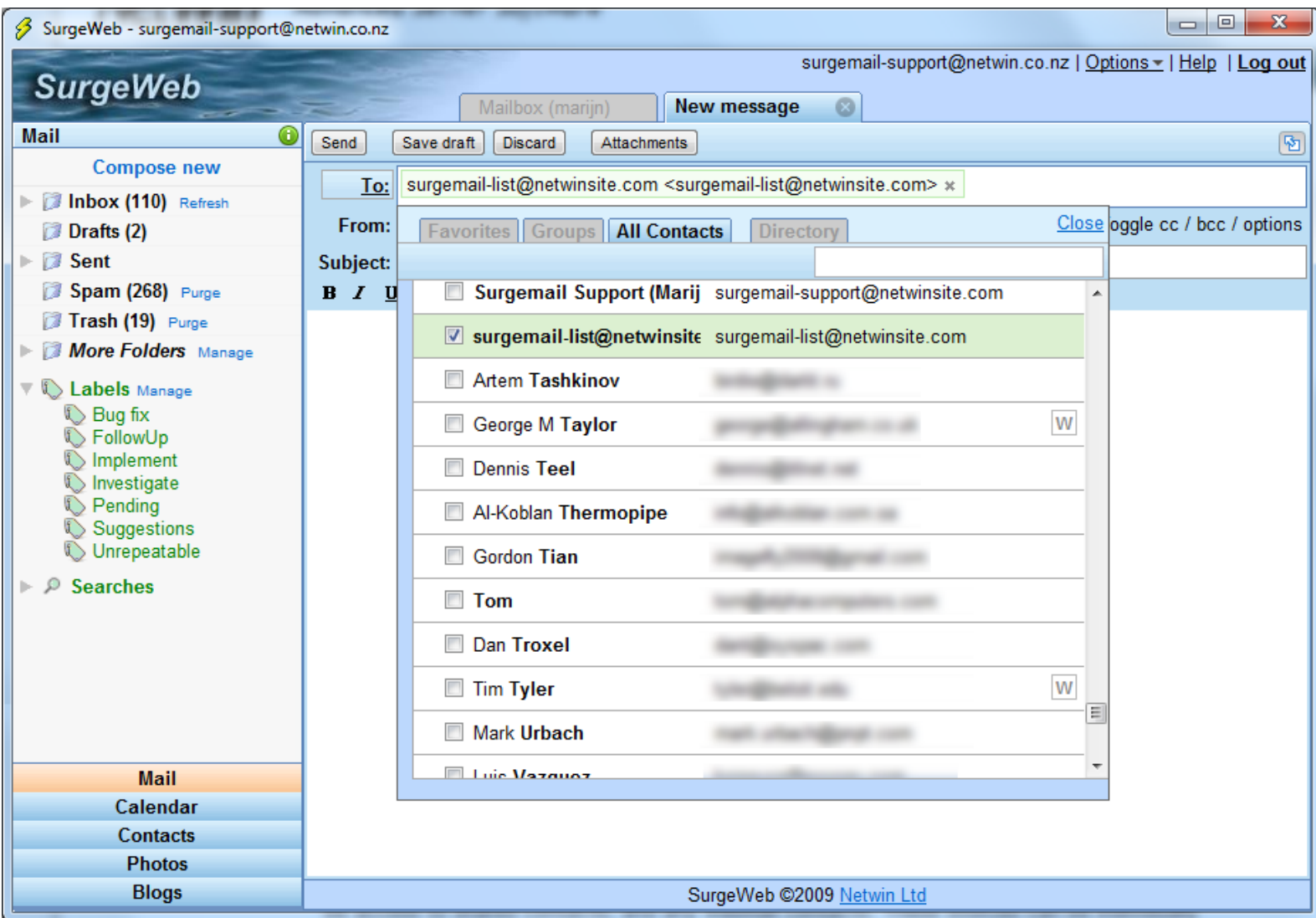
3 aa

4 aa

NOCRC_ENDMID 0 lines 0 bites 0 ctot 0

SurgeWeb ©2008 [Netwin Ltd](#)





SurgeWeb - surgemail-support@netwin.co.nz

surgemail-support@netwin.co.nz | Options | Help | Log out

SurgeWeb

Mailbox (marijn) New message

Contacts

Done New Display: Personal Server Shared Webmail More Actions

Search contact list...

all 123 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Contact Groups

Create new group

Favorites (42)

All Contacts (318)

Colleagues

Family

Friends

Netwin Staff (5)

Netwin Support (3)

Webmail default (59)

Mail

Calendar

Contacts

Photos

Blogs

SurgeMail Support

W

Surgemail Support

S

Surgemail Support (Marijn)

X

surgemail-list@netwinsite.c

X

Artem Tashkinov

X

George M Taylor

W

Dennis Teel

X

Al-Koblan Thermopipe

X

Gordon Tian

X

Tom

X

Dan Troxel

X

Tim Tyler

W

Mark Urbach

X

Luis Vazquez

X

Klaus Visser

X

Ondrej Vlcek

X

Morgan Wagner

X

webmail-support@netwin.co.nz

W

Hint: If you are trying to send an email to several contacts, then 'Compose' a new message and click on the 'To' header to bring up the address selector for your email.

This is the surgeweb contacts management page to:

- Organise your contacts into your own contact groups (ie distribution lists)

- Manage your Favorite contacts

- Search for contacts and email them

- Edit contact information

- Import / export contact information

- Edit shared addressbook information you have permission to edit

Note: Favorites is a "shortlist" of addresses containing: manually added addresses; and automatically added frequently & recently mailed addresses

SurgeWeb ©2009 Netwin Ltd

SurgeWeb

Mailbox (marijn)

Contact Groups

[Create new group](#)

Favorites (42)

All Contacts (318)

Colleagues

Family

Friends

Netwin Staff (5)

Netwin Support (3)

Webmail default (59)

Mail

Calendar

Contacts

Photos

Blogs

Contacts

Done

New

Display:

☒ Personal☒ Server Shared☒ Webmail

More Actions

Search contact list...

all

123 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Surgemail Support

S

Surgemail Support (Marijn)

x

surgemail-list@netwinsite.c

x

Artem Tashkinov

x

George M Taylor

W

Dennis Teel

x

Al-Koblan Thermopipe

x

Gordon Tian

x

Tom

x

Dan Troxel

x

Tim Tyler

W

Mark Urbach

x

Luis Vazquez

x

**Surgemail Support
(Marijn)**work surgemail-support@netwinsite.comwork 524 Kahuterawa Road,
RD4, Palmerston North,
New Zealand

Email Contact

Edit Contact

Advanced Administration

The following covers the features that administrators would wish to set up.

Generating Stat Logs

If you are using WebMail v3.0d on a linux machine you can setup WebMail to email you General stat information about your WebMail system.

The program 'webcmdlog' which is generating the stats report can only be obtained by contacting: 'support-webmail@netwinsite.com' and asking for the 'WebMail cmd.log to Stats program'. You will need to state the O/S that you are using.

An example is show below:

```
Sent: Thursday, December 25, 2003 12:00 PM
To: admin@domain.com
Subject: Webmail stats

***** Analysis of WebMail cmd.log *****
Logged period: 1 days, 11 hours, 0 mins, 33 seconds
Total CGI instances: 282095
10 Peek Seconds: CGI's / Sec      102 98 98 93 91 90 89 88 87 87
See 'results*.txt' for details for each second.

Average CGI's / Second: 4.37
Average CGI's Calls / Second: 2.24

Total Logins: 33203
Top 10 User Logins:
      19 - john@domain.com
       7 - luke@domain.com
       5 - william@domain.com
       4 - dad@domain.com
       3 - mums@domain.com
       3 - sid@domain.com
       3 - grober@domain.com
       3 - abc@domain.com
       3 - zerba@domain.com
       3 - cat@domain.com
Average Login / Minute: 15.81

Average Login Time: 1.61 seconds
10 Peek Login Times: (Seconds - User)
      11 - john@domain.com
       6 - luke@domain.com
       6 - william@domain.com
       6 - dad@domain.com
       6 - sid@domain.com
       6 - mums@domain.com
       6 - abc@domain.com
       6 - zerba@domain.com
       5 - cat@domain.com

See 'results_user.txt' for more details.

Total CGI Commands Excluding Logins: 282095
10 Most Common Commands:
  (Count) (Max Time) (Avg Time) (Part) (Max Time) (Avg Time)
58154 -      684 -      3.03 |      -      -      - reload_mail
50271 -       7 -      1.01 |      -      -      - LOGINSSCREEN
30016 -      61 -      1.99 |      -      -      - quick_login
29296 -     204 -      1.82 |      -      -      - item
23287 -      17 -      1.03 |      -      -      - menubar
12194 -     288 -      3.16 |      -      -      - delsel
10223 -      38 -      1.28 |      -      -      - logout_go
 8748 -     121 -      2.12 |      -      -      - list
 8472 -      27 -      1.13 |      -      -      - itempart
 7093 -     234 -      3.82 |      -      -      - send

10 Most Failed Commands:
  (Count) (Max Time) (Avg Time) (Part) (Max Time) (Avg Time)
See 'results_cmd.txt' for more details.

This program also created 3 or more files called:
  results_user.txt - Has the user login information
  results_cmd.txt  - Has the CGI command information
  results*.txt     - Has the time sliced in seconds

All these files are tabs seperated list designed to be imported
into spread sheets
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       372M   92M  261M  26% /
/dev/sda1        45M   14M   29M  32% /boot
/dev/sda3       1.9G  177M  1.6G  10% /home
none            125M    0  124M   0% /dev/shm
/dev/sda2       5.3G   3.2G  1.8G  63% /usr
/dev/sda7       251M  220M   17M  93% /var
```

To set this up you need to first add a webmail.ini setting:

```
enable_cmd_log true
```

Then create 2 .sh files. I suggest that you place these in a separate directory off webmail called 'cmdlog'

ie. /usr/local/webmail/cmdlog/

startlog.sh

```
rm -f /usr/local/webmail/cmdlog/cmd.log
rm results*.txt -f
/var/www/cgi-bin/webmail.cgi -ini /var/www/cgi-bin/ -enable_cmdlog /usr/local/webmail/cmdlog/cmd.log
```

stoplog.sh

```
/var/www/cgi-bin/webmail.cgi -ini /var/www/cgi-bin/ -disable_cmdlog
rm /usr/local/webmail/cmdlog/results*.txt -f
sleep 10
rm -f /usr/local/webmail/cmdlog/file.txt
/usr/local/webmail/cmdlog/webcmdlog /usr/local/webmail/cmdlog/cmd.log >/usr/local/webmail/cmdlog/file.txt
df -h >>/usr/local/webmail/cmdlog/file.txt
mail -s "Webmail stats" admin@domain.com >/usr/local/webmail/cmdlog/file.txt
rm results*.txt -f
```

You should change the directories as needed for your system, and the 'admin@domain.com' email address. The next step is setting up 2 cron jobs so that the start and stop scripts are called at the correct times. An example of this is:

CRON

```
# /etc/cron.d
0 0 * * 3 root /usr/local/webmail/cmdlog/startlog.sh
0 0 * * 4 root /usr/local/webmail/cmdlog/stoplog.sh
```

You should only run the stats over 1 day, as this does impact on the system performance when turned on.

Not only does the program provide information as above, but you can also get it to give you a breakdown of a command. To do this you run the file like this:

```
./webcmdlog cmd.log -cmd "LOGINSCREEN" >file.txt
./webcmdlog cmd.log -cmd "quick_login" >file.txt
./webcmdlog cmd.log -cmd "reload_mail" >file.txt
./webcmdlog cmd.log -cmd "logout_go" >file.txt
```

It will generate a report like this:

WebMail CMD Stat

```
***** An Analysis of WebMail 'cmd=LOGINSCREEN' *****
Total Count: 49727
Sec  Count  Percent  Total Percent
1    49712   99.97%   99.97%
2     15     0.03%   100.00%

***** An Analysis of WebMail 'cmd=attach_send' *****
Total Count: 1241
Sec  Count  Percent  Total Percent
1    758    61.08%   61.08%
2    458    36.91%   97.99%
3     15     1.21%   99.19%
4      3     0.24%   99.44%
5      3     0.24%   99.68%
6      2     0.16%   99.84%
7      2     0.16%  100.00%
```

You can also give a break down of the commands that take longer than 10 seconds, like this:

```
./webcmdlog slow.log -slow_report >> file.txt
```

It will generate a report like this:

WebMail CMD Stat

```
***** An Analysis of WebMail 'slow' *****
Total Count: 28
Count  User Path
1      d:\surgemail\web_work\u_ni\nimble@test.com\
```

```
15 d:\surgemail\web_work\u_la\largetest@test.com\  
1 d:\surgemail\web_work\u_aa\aa@test.com\  
1 d:\surgemail\web_work\u_ab\ab@test.com\  
4 d:\surgemail\web_work\u_ab\ab@test.com\  
1 d:\surgemail\web_work\u_ab\ab@test.com\  
5 d:\surgemail\web_work\u_te\test@test.com\
```

Server Farming

WebMail can support server farming, where you set up a copy of the CGI on each machine.

On one machine you set up a workarea where all the user profiles are stored.

ie. /var/spool/webmail

Then, on each of the other machines, you set up a (map/link) to this location. This way each machine can be set up to point to the one workarea location. On each machine, set the workarea ini setting to match the map/link you have setup.

With the templates you have a choice. You can either do the same as above where there is only 1 copy of the templates and each CGI points to it, or you can have a separate copy of the template set on each machine, thus saving networking bandwidth.

NOTE: If you are running the CGI on multiple machines you will need to purchase a license for each machine. Talk to sales@netwinsite.com about discounts for multiple purchases.

Auto-Login

WebMail versions 2.0f and higher have the ability to auto-login to any other NetWin products which include the auto-login feature. This saves time for the user because he/she can go from one product to another without having to login each time. The password is encrypted and stored temporarily on SurgeMail, making the process relatively safe.

In order to create links in WebMail which auto-login to another NetWin product, the following ini setting needs to be set up:

netwin_autologin_id <id> <url> <product directory> <extra>

The <id> is the id number that you wish to be set up as. ie 10

The <url> is the relative or full url to the other product. The product directory is the workarea directory of the product. WebMail also allows multiple auto-logins, so you can set up multiple ini settings like the following:

```
netwin_autologin_id 0 /cgi-bin/webnews.cgi /var/spool/webnews  
netwin_autologin_id 1 /cgi-bin/webmail.cgi /var/spool/webmail  
netwin_autologin_id 2 /cgi-bin/webmail.cgi /var/spool/webmail  
netwin_autologin_id 3 /cgi-bin/cwmail.cgi /var/spool/cwmail  
&vhost=this_host_name&tpl_set=config
```

The CGI must have write access to the 'product directory' because an encrypted password file is created which the other product uses for the login.

Once the ini setting is set up, the next step is to add a link somewhere in WebMail (on any template that logs the user into the other product). The links are as follows if the above 'netwin_autologin' ini settings are used (note: the order of the ini settings above must correspond to the numbers used in these urls):

```
<a href="||action||?cmd=netwin_login-0&utoken=||utoken||">Login to WebNews</a>  
<a href="||action||?cmd=netwin_login-1&utoken=||utoken||">Login to WebMail</a>  
<a href="||action||?cmd=netwin_login-2&utoken=||utoken||">Login to WebMail</a>  
<a href="||action||?cmd=netwin_login-3&utoken=||utoken||">Login to CWMail</a>
```

NOTE: This will only work if the username and password are the same for both products.

Below are the common autologin setups in WebMail and how to set them up. Any ini setting changes stated will also have next to them the ini file name brackets like '(4th Setting)' What this means is that this is the autologin setting position it is expected to be in. For example the '4th setting' means that there is expected to already be 3 other autologin ini settings in the ini file before these settings. The above only matter for (netwin_autologin and not netwin_autologin_id)

If the order of the autologins do change, then WebMail logins will not login to the expected product. If you wish to change the order OR the order has to be different due to already setup settings then you will need to also change the templates commands to also match this.

Note: The versions stated below in each, are the version you need to get ALL of the feature stated. Older versions of

WebMail will have some of the below, but some features will not be available without extra template changes.

[WebMail <--> SurgeMail](#)

You need to have **WebMail v3.1a** and **SurgeMail v1.6h** product version for this to work as there are template changes that are already included in these versions. For simple addition of this feature. **SurgeMail** defaults to use these settings and normally no changes are needed unless you are upgrading from an earlier version. If you are setting up **WebMail** on a separate machine to that of **SurgeMail**, or using another web server, you will need to do some of the changes below as well.

What this interface does:

- In WebMail, on the user configuration page they can:
 - Change their mail password.
 - Setup Server Spam Settings.
 - Setup Holiday Settings
 - Centerpaid Settings
 - SMS Settings
 - Forwarding Settings and others
- In SurgeMail the users can move to WebMail without having to login.

Setup:

As long as you have the stated version above or higher, all you should need to do is add a few ini settings in WebMail and the interface should work for the default installations. No changes to SurgeMail are needed as it will default to having these settings setup.

WebMail.ini

```
friends_only true
autorespond true

netwin_autologin_id 0 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_pass&||na_extra||
netwin_autologin_id 1 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_fcommon&||na_extra||
netwin_autologin_id 2 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_fwd&||na_extra||
netwin_autologin_id 4 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_spam_load&||na_extra||
netwin_autologin_id 5 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_centipaid&||na_extra||
netwin_autologin_id 6 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_sms_load&||na_extra||
netwin_autologin_id 7 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_spam_load&||na_extra||
netwin_autologin_id 8 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_listmb&||na_extra||
netwin_autologin_id 20 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_pass&vhost=||vhost||
netwin_autologin_id 21 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_fcommon&vhost=||vhost||
netwin_autologin_id 22 http://1.2.3.4:7080/cgi/user.cgi
                        C:\surgemail\work lcmd=user_load_fwd&vhost=||vhost||
```

[WebMail <--> DBabble](#)

You need to have **WebMail v3.1a** and **DBabble v1.1m** product version for this to work.

What this interface does:

- When you are in WebMail you can quickly move to DBabble web based session sending instant messages to friends and work mates.
- When you are in DBabble web based session you can quickly move to WebMail to read and send emails.

Setup:

For this you need to add 1 ini setting and 1 template line in each product so that you can move between the products.

WebMail.ini (10th Place)

```
netwin_autologin_id 10 http://1.2.3.4:8132
C:\dbabblesvr\work
```

WebMail Template Addition

```
<a href="||action||?cmd=netwin_login-10&utoken=||utoken||">Check DBabble</a>
```

DBabble.ini (10th Place)

```
auto_login_user nobody
auto_login 10 http://$(server)/scripts/webmail.exe
c:\webmail
```

DBabble Template Addition

```
<a target="||u_top||" href="||action||?
cmd=send_auto_login&n=10&tok=||tok||">Read Mail</a>
```

[WebMail <--> WebNews](#)

You need to have **WebMail v3.1a** and **WebNews v1.1o** product version for this to work.

What this interface does:

- When you are in WebMail you can quickly move to WebNews read your news articles without having to login to WebNews.
- When you are in WebNews you can quickly move to WebMail read your emails, send a few out without having to login to WebMail.

Setup:

For this you need to add 1 ini setting and 1 template line in each product so that you can move between the products.

WebMail.ini (200th Place)

```
netwin_autologin_id 200 /scripts/webnews.exe
C:\webnews
```

WebMail Template Addition

```
<a href="||action||?cmd=netwin_login-200&utoken=||utoken||">Read News</a>
```

WebNews.ini (varies)

```
netwin_autologin /scripts/webmail.exe C:\webmail
```

WebNews Template Addition

```
<a href="||action||?cmd=netwin_login-1&utoken=||utoken||">Read Mail</a>
```

[WebMail --> NetAuth](#)

You need to have **WebMail v3.1a** and **Netauth v4.2I** product version for this to work as their are template changes that are already included in these versions. (**Note:** See [WebMail <--> SurgeMail](#) if you are using Surgemail as your mail server)

What this interface does:

- When you are in WebMail and you move to the configuration page you can select to 'Change Your Password' for the main mail account. This calls NetAuth which is will do the change and then return back to WebMail.

Setup:

NetAuth has been out-dated due to SurgeMail, so WebMail templates do not have the NetAuth links and need to be manually added.

WebMail.ini (11th Place)

```
netwin_autologin_id 11 /scripts/netauth.exe C:\netauth
```

&show=passwd.tpl

Config Template Change

In one of the config templates you will need to add this link:
Change Password

[WebMail <--> WebShareIt](#)

You need to have **WebMail v3.0u build 42** and **WebShareIt v1.0j** product version for this to work as their are template changes that are already included in these versions, for simple addition of this feature.

What this interface does:

- When you are in webmail and you are writing an email and you wish to attached a file that is stored in WebShareIt you can just click on the 'webshareit' link that will appear on the attachment page where normally you attach a file. It will popup a window which will login to webshareit and then allow the you to select 1 file that they wish to attach. Which is then passed back to webmail, allowing the user to then send the email, and the file will be attached.
- When you are in WebShareIt a new button called 'Email' will appear when you are looking though your folders. If you select 1 file and then clcik this button it will popup window which will login to webmail open a new message and attached the selected file. So the user can continue writing the email and send it off.

Setup:

As long as you have the stated version above or higher all you should need to do is add 4 ini settings, 2 in WebMail and 2 in WebShareIt and the interface should work for the default installations.

WebMail.ini (12th Place)

```
display_webshareit c:\webshareit\users
netwin_autologin_id 12 /scripts/webshareit.exe
C:\webshareit\users
show=webmail.tpl&path=||_webshare_path||
```

WebShareIt.ini (1st Setting)

```
display_email true
autologin /scripts/webimap.exe d:\webmail
&process=show&page=framenew&
listcmd=share_file&file_path=||send_file_path||
```

[WebMail <--> External Program](#) or Single Sign On, SSO

The below explains what an external program mst to do autologin to WebMail.

You need to have atleast **WebMail v3.0u build 42** and **SurgeMail v1.6h** product version for this to work.

External Program --> WebMail

- First the external program MUST have access to the username and password of the account you wish to login with.
- The program encodes the user/pass using the following code:

```
encoded_pass = pass_user_decrypt(pass,user);
char *pass_user_encrypt(char *pass, char *user)
{
    if (user && pass) {
        static char enc[BFSZ];
        char *u, *p;

        lcpy(enc, pass, BFSZ);
        for (u=user, p=enc; *p; p++) {
            *p += *u++;
            if (!*u) u = user;
        }
        return enc;
    }
    return NULL;
}
```

- The external program then opens a SurgeMail's POP port and send the encoded pass like this:

```
C: putp encoded_pass
S: +ok id
```


The 'id' that the SurgeMail must be passed on to WebMail.

- The last step is calling WebMail with the following information:

?cmd=auto_login&user=user&id=id

External Program <-- WebMail

- Once you have WebMail setup to have the correct webmail.ini settings to call the autologin to the external program. (see above for settings). WebMail will call the external program like this:

?cmd=auto_login&user=user&id=id

- The program then need to grab the 'id' section and opens a SurgeMail's POP port and send the ID like this to get the encoded_pass:

C: getp id
S: +ok encoded_pass

- The external program can then decode the encoded_pass using the function below and then verify the user/pass to ensure that the login is allowed.

```
pass = pass_user_decrypt(encoded_pass,user);
char *pass_user_decrypt(char *encoded, char *user)
{
    if (user && encoded) {
        static char enc[BFSZ];
        char *u, *p;

        lcpy(enc, encoded, BFSZ);
        for (u=user, p=enc; *p; p++) {
            *p -= *u++;
            if (!*u) u = user;
        }
        return enc;
    }
    return NULL;
}
```

Auto-Login (Without SurgeMail)

WebMail versions 3.0u and higher have the ability to auto-login to any other NetWin products which include the auto-login feature. This saves time for the user because he/she can go from one product to another without having to login each time. The password is encrypted and saved in a temporary file on the machine (not accessible from the net), making the process relatively safe.

Since that you are not using our mail server (SurgeMail) you have to use the old method which has a limiting factor.

"The WebNews and WebMail products MUST be able to get access to each other workarea's."

The reason for this is that the password is encoded and stored in a file which the other product then grabs and uses.

This means that the programs need to be on the same machine or that you have to setup network drives so that each can store files in the correct location. In this case you have to be careful of ownership of files that each will create as both will need to read/write and delete the files.

[WebMail <--> WebNews](#)

You need to have **WebMail v3.1a** and **WebNews v1.1o** product version for this to work.

What this interface does:

- When you are in WebMail you can quickly move to WebNews read your news articles without having to login to WebNews.
- When you are in WebNews you can quickly move to WebMail read your emails, send a few out without having to login to WebMail.

Setup:

For this you need to add 1 ini setting and 1 template line in each product so that you can move between the products.

WebMail.ini (200th Place)

```
netwin_autologin_id 200 /scripts/webnews.exe
C:\webnews
```

WebMail Template Addition

```
<a href="||action||?cmd=netwin_login-9&utoken=||utoken||">Read News</a>
```

WebNews.ini (varies)

```
netwin_autologin /scripts/webmail.exe C:\webmail
```

WebNews Template Addition

```
<a href="||action||?cmd=netwin_login-1&utoken=||utoken||">Read Mail</a>
```

[WebMail <--> External Program](#)

The below explains what an external program must do to autologin to WebMail.

You need to have atleast **WebMail v3.0u build 42** for this to work.

External Program --> WebMail

- First the external program MUST have access to the username and password of the account you wish to login with.
- The program encodes the user/pass using the following code:

```
encoded_pass = pass_user_decrypt(pass,user);
char *pass_user_encrypt(char *pass, char *user)
{
    if (user && pass) {
        static char enc[BFSZ];
        char *u, *p;

        lcpy(enc, pass, BFSZ);
        for (u=user, p=enc; *p; p++) {
            *p += *u++;
            if (!*u) u = user;
        }
        return enc;
    }
    return NULL;
}
```

- The external program then opens a file in the workarea of WebMail and save the encoded_pass in this file. The file MUST end with '.tmp'
- The last step is calling WebMail with the following information:

```
?cmd=auto_login&user=user&pass_file=file
```

The file must NOT have the file extension (.tmp) on it.

External Program <-- WebMail

- Once you have WebMail setup to have the correct webmail.ini settings to call the autologin to the external program. (see above for settings). WebMail will call the external program like this:

```
?cmd=auto_login&user=user&pass_file=file
```

- The program then need to grab the 'file' open it to get the encoded_pass. The program should then remove this file.
- The external program can then decode the encoded_pass using the function below and then verify the user/pass to ensure that the login is allowed.

```
pass = pass_user_decrypt(encoded_pass,user);
char *pass_user_decrypt(char *encoded, char *user)
{
    if (user && encoded) {
        static char enc[BFSZ];
        char *u, *p;

        lcpy(enc, encoded, BFSZ);
        for (u=user, p=enc; *p; p++) {
            *p -= *u++;
            if (!*u) u = user;
        }
        return enc;
    }
    return NULL;
}
```

WebMail Command Prompt Options

WebMail has a few built in commands that you can run via command prompt. The list of the available commands in your current version of WebMail can be view by running WebMail like this:

```
./webmail.cgi -h
or ./webmail.cgi -?
```

This will then list all the available command like this:

```

Welcome to WebMail v3.1m

Layout: webmail -version
        webmail -activate registration_number email_address
        webmail -deactivate registration_number email_address
        webmail -password <password>
        webmail -lang_tpl
-activate | Displays version Information
-activate | This will activate a webmail key
-deactivate | This will deactivate a webmail key
-password | This will set the managers password. ini setting si ignored
-lang_tpl | This will get webmail to rebuild the language tpl from
           | the mastersets.

Layout: webmail -admin_fns
        webmail -admin_fns_full
        webmail -manager
        webmail -deleteuser <password> wild_user [wild_user [...]]
        webmail -checkutoken <user token>
        webmail -check <user@domain>
        webmail -show_path <user@domain>
-admin_fns | This will force the CGI to check for
           | 'auto_delete wild xdays [ydays]'
           | and clear only logged in users (login.dat).
-admin_fns_full | This will force the CGI to check for
           | 'auto_delete wild xdays [ydays]'
           | and clear ALL the users caches.
-manager | Command line managers control
-stats | This will geneate the manager login reports
-deleteuser | Command line delete user lists
-checkutoken | Check that the user token is currently valid
-check | This checks if a user account is active (login session)
-show_path | This will return back the full path to the user

Layout: webmail -stats
        webmail -space
        webmail -clean)
        webmail -space_pop
        webmail -clean_pop
        webmail -enable_cmdlog
        webmail -disable_cmdlog
These are more advance setting and should use with extreme care.

Layout: webmail -remove_lock <username>@<domain>
        webmail -test_lock <username>@<domain> [<sleep in seconds>]
-remove_lock | This attempts to remove the lock files for a user.
-test_lock | This will test the locking routines for a user.
```

The '-stats' command is covered in the section '[Setup Webmail to generate more stat information](#)' and will tell you how to setup and use this feature. Most of the others are self explanatory in their comments. I am going to talk about the '-manager' option.

The '-admin_fns' command cycle thoughts the login users (login.dat) and check to see if they should be logout and clears their cache at the same time. We suggest that you should set this up as a cron job to be run once every hour, and also including the webmail.ini setting:

```
command_admin_only true
```

This will stop the CGI from processing this command which for large systems can cause large delays once in a while.

The '-manager' is the command prompt manager screen with similar features to that of the web base manager. But were the web base can timeout for large system the command prompt does not have this issue. Once you run the manager option you will be asked for the managers password that you must have setup in the webmail.ini file before hand. Then CGI will then load all the user information in removing and empty users and display a menu like the following:

```

Welcome To Managers Page
-----

Stats:   Located xxx Users based in d:\webimap\

1. Display selected users to screen
2. Delete selected user
3. Change User.dat Variable for selected users.
```

- d. Delete all old Accounts. (ini setting - auto_delete)
- c. Clear All Users Caches. (ini setting - auto_logout)
- f. Delete Users from File.
- s. Switch user directories from one pop host to another.

- i. Users Information. (Displays how long since they lasted logged in)
- u. Delete Users Un-used for x days.

- r. All the email in user Y's mailbox older than X days are removed. (POP only)
- e. All the emails in user Y's folder X are removed. (POP only)
- v. Verify all user.dat files - user.dat repairs.
- x. Update users quotas (POP folders only).

- m. Move POP folder to IMAP server.

- q. exit

The number of users on your system will be shown and the list of available options. Each of these options are explained below:

1. Display selected users to screen

If you choose this option you will be asked for a wild card list of the users you wish to select. If you enter: 'lynden@*' this will display all the 'lynden' users of every domain that has used this CGI.
You will also be given the option to save this list to a file

2. Delete selected user

If you choose this option you will be asked for a wild card list of the users you wish to select. If you enter: 'lynden@*' this will display all the 'lynden' users of every domain to the screen. You will then be asked if you are sure you wish to remove these users.

3. Change User.dat variables for selected users.

If you choose this option you will be asked for a wild card list of the users you wish to select. If you enter: 'lynden@*' this will display all the 'lynden' users of every domain to the screen. You will then be asked what variable name and what value you wish to change it to. You will be provide a list of common user.dat variables to select from.

d. Delete all old Accounts. (ini setting - auto_delete)

If you have already setup 'auto_delete' ini setting this option will cycle through all your users and delete any accounts that need to be removed.
This is normally only used if you just changed the ini settings and wish then to process straight away.

c. Clear All Users Caches. (ini setting - auto_logout)

If you have already setup 'auto_logout' ini setting this option will cycle through all your users and logout the required accounts that match you ini settings.
This is normally only used if you just changed the ini settings and wish then to process straight away.

f. Delete Users from File.

This option allows you to import an external file which lists (one on each line), the users you wish to remove. This option will login this file, display the list of users and ask if you are you sure you wish to remove these accounts.

s. Switch user directories from one pop host to another.

If you are moving your users to another POP/IMAP server or even a different domain, you will need you use this feature. This will convert all the users in webmail to the new POP host. You should use this option with care. If this is not used correctly, your users might lose their settings.

i. Users Information. (Displays how long since they lasted logged in)

If you choose this option you will be ask for a wild card list of the users you wish to select. If you enter: 'lynden@*' this will display all the 'lynden' users of every domain that has used this CGI.
This will then tell you when they last logged in and give you the option to save this to a file.

u. Delete Users Un-used for x days.

This will allow you to delete users depending on when they last used WebMail. You will display the list of users and be asked if you are sure that you wish to delete these accounts.

r. All the email in user Y's mailbox older than X days are removed.

This option only works for POP folders, allowing the admin to remove emails that have been there for some time.

e. All the emails in user Y's folder X are removed. (POP only).

This option only works for POP folders, allowing the admin to remove an entire folder for a group of users.

v. Verify all user.dat files - user.dat repairs.

This will cycle through all your users and verifying the user.dat and repairing any damage that past CGI might have caused. You will shown a list of user.dat that have been repaired.

x. Update users quotas (POP folders only).

This will cycle through all your users and rebuild all the POP folders to ensure that the POP quota for these users are correct. It will also fix any index.dat issues.

m. Move POP folder to IMAP server.

This feature allows you to move all the POP folders stored on the web server, to be uploaded to the IMAP server. This is only used if you are moving all your users to your IMAP server, and to move the POP folders as well.

Using Gnu Privacy Guard (GnuPG)

If you are using WebMail v3.0u or higher you can set it up to use GnuPG. For more information about this product see the URL:

<http://www.gnupg.com/>

The software for safe and encrypted e-mail-communication. GnuPG is published under General Public License (GPL) and there is a free software alternative to Pretty Good Privacy (TM), shortly known as PGP (TM). GnuPG is based on OpenPGP-standard.

To setup WebMail with GnuPG you first need to download GnuPG, compile and install in on your system. Due to the way GnuPG interfaces with the console, changes are needed to GnuPG to made so that webmail can correctly run gpg correctly. So before you build GnuPG you will need to replace './util/ttyio.c' with the one provided [here](#).

Windows and Linux Libc6 compiled version can be downloaded from the table below.

Windows (gpg.exe)
Linux Libc6 (gpg_netwin)

This is just the gpg.exe you will still need to download the orginal GnuPG and install it.

Once install you need to add to webmail.ini file the following:

pgp_path c:\pgp\gpg.exe
or pgp_path /usr/local/bin/gpg_netwin

This will then activate webmail pgp code and on the Panel tpl set on the 'Option' page. This new option called:

PGP Profile

This will give you the option to generate a new Public/Private key, which is then used when sending emails. This will also list teh current Key that other users have sent you. On the 'New Message' page at teh top a new pull down menu will appear giving you the option to send your key or sign the message using yor key.

Warning: Since GnuPG is 3rd party software, Netwin Ltd is not liable for any damage caused by the use of this software, or any licensing requirements.

Generate more Stat Information

WebMail v3.0t and higher has more stat information that you can collect and use. The main information that customers want are:

How many emails were sent this month.
How many logins occured this month.

- Also A break down of how many times each user login.
How many times a template page was displayed. (ie login.tpl, options page etc)

To set up the above you need to add the following ini settings:

```
# How many emails were sent this month.  
log_sent_emails true  
  
# How many logins occurred this month. (Including break down)  
log_login_users true  
  
# How many times a template page was displayed  
log_template item.tpl view_email  
log_template login.tpl login_page
```

The stat information is display on the managers page at the URL like the following:

```
http://your.domain.com/cgi-bin/webmail.cgi?cmd=manager
```

All this information is stored in WebMail's workarea, each in a seperate file. You might need to manually remove old files if they start taking up to much room.

SurgeWeb - marijn@netwin

192.168.1.100:7026/surgeweb#

☆ 🔍

marijn@netwin.co.nz | Options | Help | Log out

SurgeWeb

Mailbox (Inbox)

Calendar

Today New Event

Day Week Month Year

26 27 28 29 30 31 1

2 3 4 5 6 7 8

9 10 11 12 13 14 15

16 17 18 19 20 21 22

23 24 25 26 27 28 29

<< < > >>

All None Configure

☒ Calendar

☒ Calendar 2

☒ Tentative

Shared Calendars

☒ Calendar | Chrisp (R)

☒ Calendar | Ralph (RW)

Delivery log, Spam control

Mail

Calendar

Contacts

Photos

Blogs

Sep 2012

all day

2 Sunday

3 Monday

4 Tuesday

5 Wednesday

6 Thursday

7 Friday

8 Saturday

Amy's Brithday

Long weekend

9 AM

10 AM

11 AM

12 PM

1 PM

2 PM

3 PM

4 PM

5 PM

6 PM

7 PM

9 – 10

9 – 10

9:30 – 10:30

10:30 – 12p

1p – 2p

1:30p – 2:30p

3p – 5p

6p – 7p

9 – 10

9 – 10

10:30 – 11:30

12p – 1p

1p – 2p

2p – 3p

3p – 4p

3p – 4p

3p – 4p

3p – 6p

6p – 7p

6p – 7p

6p – 7p

6p – 7p

6p – 7p

Veterinarian

Server check

Car service

Visit John

Cal2 Evt1

Cal2 Evt2

Cal2 Evt 3

Daily evenings!

Daily evenings!

Daily evenings!

Daily evenings!

Daily evenings!

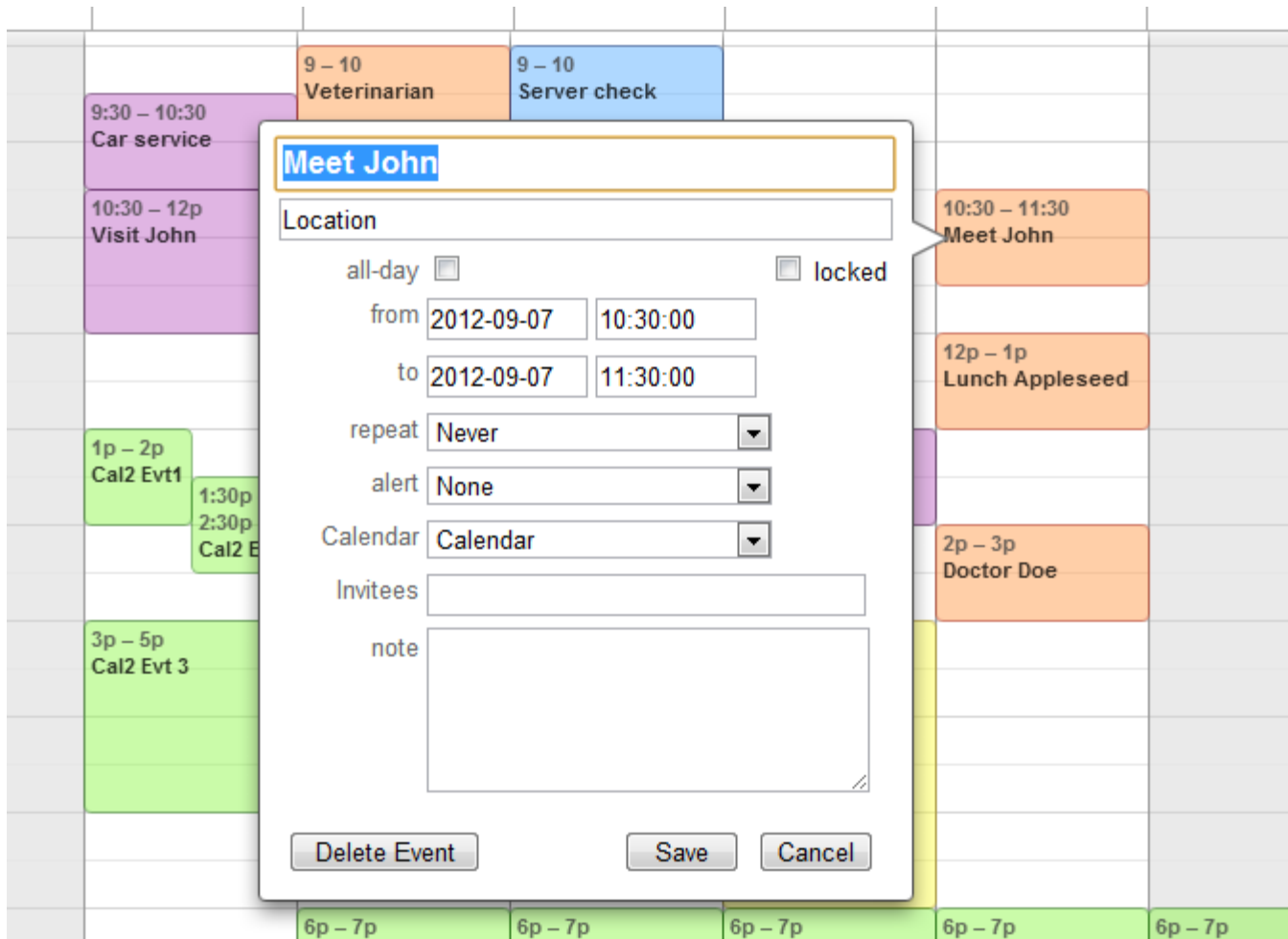
Meeti

Meeti

Meeti

Maybe meet Y

SurgeWeb ©2009-2012 Netwin Ltd



SurgeWeb - marijn@netwin

192.168.1.100:7026/surgeweb#

☆ 🔍

marijn@netwin.co.nz | Options | Help | Log out

SurgeWeb

Mailbox (Inbox)

Calendar

Today New Event

Day Week Month Year

Jan Feb Mar Apr

May Jun Jul Aug

Sep Oct Nov Dec

<< >>

All None Configure

☒ Calendar

☒ Calendar 2

☒ Tentative

Shared Calendars

☒ Calendar | Chrisp (R)

☒ Calendar | Ralph (RW)

Delivery log, Spam control

Mail

Calendar

Contacts

Photos

Blogs

Sep 2012

9:00 am busy1
10:00 am busy2
11:00 am busy3
2:00 pm busy4
2:30 pm busy5
3:00 pm busy6

SurgeWeb ©2009-2012 Netwin Ltd

SurgeWeb - marijn@netwin

192.168.1.100:7026/surgeweb#

☆ 🔑

SurgeWeb

marijn@netwin.co.nz | Options | Help | Log out

Mailbox (Inbox)

Calendar

JanFebMarApr

MayJunJulAug

SepOctNovDec

<<<>>>

AllNoneConfigure

☒ Calendar

☒ Calendar 2

☒ Tentative

Shared Calendars

☒ Calendar | Chrisp (R)

☒ Calendar | Ralph (RW)

Calendar Settings

Done

CalDAV calendars

Standalone CalDAV based calendaring for mobile and desktop clients. You can create additional personal calendars direct from your mobile device (eg iPhone). To edit calendar sharing permissions with other email accounts you will need to use this interface.

Create new calendar:

Create

Your Calendars

Name	Color	Shared With	Actions
Calendar	<div></div>		sharing del copy url
Calendar 2	<div></div>		sharing del copy url
Tentative	<div></div>		sharing del copy url

Shared Calendars

Name	Color	Owner	Actions
Calendar Chrisp (R)	<div></div>	Shared by chrisp@netwin.co.nz with read only access	copy url
Calendar Ralph (RW)	<div></div>	Shared by ralph@netwin.co.nz with read / write access	copy url

Your iOS devices should connect to your calendar by just filling our username@your.domain, password, server name. If other CalDAV clients need the fully specified url this is "http://your.server/cal/calendars/user@your.domain/" ([more documentation](#)).

Location:

Pacific/Auckland

(make sure this is set correctly)

Prevent drag & drop editing of:

All can be edited

(prevents accidental editing)

Save Settings

SurgePlus calendar

The standalone surgeplus calendar is still available. Switch back to using surgeplus calendar [for this session](#).

Delivery log, Spam control

Mail

Calendar

Contacts

Photos

Blogs

SurgeWeb ©2009-2012 Netwin Ltd