Sun Internet Mail Server™ 4.0 Product Update

This product update reflects changes made to the SIMS 4.0 release introduced by the 06 revision of the SIMS product patch (108049-06 for SPARC and 108050-06 for Intel). The product update notes cover installation and software bugs for SIMS 4.0. The patch update process is outlined as follows:

- Read through this Product Update.
- Obtain the software patches from SunSolve.
- Install the new software patch on the system running SIMS 4.0.

SIMS 4.0 includes Sun WebAccess. If you are using Sun WebAccess, the required patches are WebAccess patch 108207-03 (SPARC) or 108208-03 (Intel).

Product Update Notes.

Items covered in this product update are cumulative from the initial release of SIMS 4.0, but items that have been fixed in previous patch releases may have been dropped from release notes. SIMS patches are cumulative. For a complete list of bug fixes, please refer to the patch readme file.

Installation Updates

This section describes updates and workarounds for the known problems that occur during installation and initial configuration.

Installation Fails with root umask Set to 227 (bug 4270752)

Description: An address error occurs during SIMS 4.0 installation. Because the root umask is set to 227 (read and execute permission only for owner and group), the IMTA configuration file could not be properly created.

Workaround: Before installation, set the root umask to 227 (read, write, and execute permission for owner; read and execute permission for group and other).

Disappearing Lines in slapd.conf (bug 4250205)

Description: Bringing up the SIMS Admin Console with the Netscape browser causes the schema files to disappear from the `slapd.conf` file. The console rewrites the `slapd.conf` file, removing the following lines:

```
include "/usr/netscape/server4/slapd-<host>/config/sims-sisp.at.conf"
include "/usr/netscape/server4/slapd-<host>/config/sims-sisp.oc.conf"
include "/usr/netscape/server4/slapd-<host>/config/sims.at.conf"
include "/usr/netscape/server4/slapd-<host>/config/sims.oc.conf"
```

The old file is renamed `slapd.conf.old`.

Workaround: Restart slapd before starting the SIMS Admin Console:

```
# /usr/netscape/server4/slapd-<hostname>/stop-slapd
# /usr/netscape/server4/slapd-<hostname>/start-slapd
```

Fatal Error During Install (bug 4248148)

An error results when installing SIMS 4.0 using `setup-tty` with the `sims_setup.dat` file and the following options: Web Access, SDK, SDK documentation, SIMS 4.0 documentation, varmail, Sun Directory Services, remote LDAP host, and alternate LDAP Port.

You will receive an error similar to:

```
Jun 21 10:50:02 slim SUNWmail.ims.imta_dirsync[6399]: Cannot open
directory/file /etc/opt/SUNWmail/ims/ims.cnf: No such file or
directory

Jun 21 10:50:02 slim SUNWmail.ims.imta_dirsync[6399]: Fatal error
```

Note – After the error, installation proceeds and finishes successfully.

A dirsync cron job is set up by the postinstall script of SUNWimimo package. This job is set to execute at 10, 30 and 50 minutes every hour. The message store configuration file `ims.cnf` is created towards the end of install during the configuration phase. If the cronjob executes between the time the cronjobs are set up and the message store configuration file is created (usually between 8-10 minutes), this error will result.

Apart from the display on the console, this is not fatal and the next time the cron job is executed, it will succeed.

Some Directories Remain After Uninstall (bug 4267591)

After uninstalling SIMS 4.0, certain files associated with Web Access and Sun DS are not removed; hence, their directories are not removed. These files are co-packaged with SIMS 4.0 but are not really a part of the SIMS product. They reside in the following directories:

- `/var/opt/SUNWconn`
- `/opt/SUNWconn`
- `/opt/SUNWa`

You can remove these files and directories with the `rm -rf` command.

NSDS Must Not Be Running When Importing Schema (bug 4317165)

Pages 130-131 of the SIMS 4.0 Install Guide (Appendix A) gives the process for adding the schema for SIMS into Netscape Directory Server. It does not tell you to stop `ns-slapd`. If `ns-slapd` is running while you are doing this, the files that you have modified will be copied back from the `conf_bk`, and your changed files will be saved with a `.old` extension.

Stop Netscape Directory Server when changing its configuration files.

Installing SIMS with SunCluster Patch

SunCluster Patch will not work on systems with SIMS 4.0 with HA and the Netscape Directory Service (NSDS) unless the modifications described in this note are made to the SIMS and SunCluster configuration. The current patch can be downloaded from:

`http://sunsolve.sun.com`

Advantage of Applying SunCluster Patch

This patch contains a fix that enables the Netscape Directory Service probing feature. This feature allows the HA server to probe for the `slapd` process. In an event of `slapd` failure, the HA server will try to restart the service. If restart attempts fail, the service will be restarted on the backup node. This provides more directory service reliability.

▼ To Configure the SunCluster

These instructions describe the procedures for new SunCluster installations as well as how to modify existing installations. These instructions make extensive references to “Guidelines for Installing and Configuring SunCluster and High Availability,” Page 147 of the SIMS 4.0 Installation Guide.

1. **Install this patch on both cluster nodes.**

2. **Run** `/opt/SUNWcluster/bin/hadsconfig` **on both cluster nodes.**

   Use “Create instance” if this is the first time running `hadsconfig`. Use “Edit instance” if you’ve already run `hadsconfig`.

   Next to **Name of the instance:** enter `nsldap`

   Next to **Base directory of product installation:** enter `/<shared-file-system>/NSDS/slapd-<ha-logical-hostname>`

   Use default values for other parameters of `hadsconfig`, and save the changes. These parameters are shown in step 5 of “Guidelines for Installing and Configuring SunCluster and High Availability.”

3. **If you have completed step 6 of** “Guidelines for Installing and Configuring SunCluster and High Availability” **in an earlier installation, undo the changes on both cluster nodes.**

   That is, make sure the line `method_timeout='hareg -q nsldap -T stop'` exists in `/opt/SUNWcluster/ha/nsldap/nsldap_svc_stop`

   Skip step 7 of “Guidelines for Installing and Configuring SunCluster and High Availability”.

4. **Re-register and re-start both `nsldap` and SIMS data services.**

   Refer to page 152 of the SIMS 4.0 Installation Guide on how to register the Netscape directory service with the High Availability Framework. (If the data services have been registered, they need to be unregistered prior to being re-registered.) Replace steps 3-5 as follows:

   3. Register the NSDS/HA service:

      ```
      # /opt/SUNWhadf/bin/hareg -s -r nsldap
      ```

   4. Start the NSDS/HA service:

      ```
      # /opt/SUNWhadf/bin/hareg -y nsldap
      ```

   5. Re-register the SIMS/HA service:

      ```
      # /opt/SUNWhadf/bin/hareg -r Sun_Internet_Mail -b /opt/
      SUNWimha/clust_progs -m START_NET=imha_start_net,
      STOP_NET=imha_stop_net -t START_NET=120,STOP_NET=30 -v 4.0 -d
      nsldap
      ```

Procedure for Removing the NSDS Data Service

Refer to page 155 of the SIMS 4.0 Installation Guide on how to remove the NSDS Data Service. Replace steps 3-4 with the following:

3. Stop the NSDS/HA service:

```
# /opt/SUNWhadf/bin/hareg -n nsldap
```

4. Unregister the NSDS/HA service:

```
# /opt/SUNWhadf/bin/hareg -u nsldap
```

Software Updates

This section describes known software limitations that are fixed and software updates that are introduced by this revision of the SIMS 4.0 product patch or are still open from previous patch versions.

Some Messages on Logging File System Were Lost When Panic Occurred (bug 4333001)

Users cannot access a portion of sent mail after the message store server was stopped during a panic.

Workaround: The problem is partially fixed in SIMS 4.0 patch 06. After installing SIMS 4.0 patch 6, you must run `imcheck -t` before restarting the server after a crash.

Simple Family Account Implementation (bug 4280093)

Simple Family Accounts enhancements have been added to the Delegated Management Console.

Through the DM Console, an administrator has the ability to:

- Create, modify, or delete a Family Account
- Define or modify a family head
- Define or modify the services for each Family Account
- Define or modify the maximum family size
- Define or modify the maximum family quota
- Generate a Family Account report

Also through the DM Console, a family head has the ability to:

- Create, modify, or delete a family member
- Define or modify the services for each family member
- Generate a Family Account report

A set of command line interfaces has also been included. Through the CLI, a family head can add, modify, or delete a family member, and generate a Family Account report.

Korean to UTF-8 iconv Module is not Found (bug 4262729)

The Korean to UTF-8 iconv module is not found when called by name.

This bug has been partially fixed in the patch 05 revision. The following links should be made in `/usr/iconv/lib/` during install or as a workaround:

```
# ln -s ko_KR-iso2022-7%ko_KR-UTF-8.so ko_KR-iso2022-7%UTF-8.so
# ln -s ko_KR-euc%ko_KR-UTF-8.so ko_KR-euc%UTF-8.so
```

The Site Admin's Password is Revealed when `imdelegatedmd` is Run From the Command Line (bug 4264225)

When `imdelegatedmd` is run from the command line, a line similar to the following appears in the output, revealing the `siteadmin`'s password:

```
: LDAP_DN_PASSWORD =                secret
```

Although this bug was fixed in SIMS 4.0 patch 4, the bug will remain in release notes as a security issue. If the fix is important for your site, it is recommended that you upgrade to SIMS 4.0 patch 4 or higher.

imaccessd Allows the User to View the Entire Disk (bug 4258547)

If a user is created with `/var/mail` instead of the mail server, the user can telnet to the server and view the entire system with the following command:

```
list / *
```

Although this bug was fixed in SIMS 4.0 patch 4, the bug will remain in release notes as a security issue. If the fix is important for your site, it is recommended that you upgrade to SIMS 4.0 patch 4 or higher.

Domain Quota Limits are not Enforced (bug 4295401)

Domain Quota Limits are not enforced in this version of the product due to the following technical issues and performance trade-offs:

- The ldap server does not support transaction locking; therefore, the quota counts are subject to race conditions and may not be accurate.
- The operation of the Delegated Management (DM) Server is hampered. When a user logs in, the DM counts the number of users and distribution lists in the directory by searching for all of them. If there are a lot of users, this can take a considerable amount of time. Additionally, as the number of users approaches the quota limit, the DM recounts the number of users.

The imta dirsync Takes a Long Time When Migrating from SIMS 3.5 to 4.0 (bug 4279420)

The `lookthroughlimit` attribute cannot handle a value of "-1"; if this attribute is set to "-1", the indices will be skipped. By default, this value should be set to at least 50,000. This attribute is located in the `slapd.ldbm.conf` file.

Additionally, there is an existing bug with case sensitivity of the attribute names in `slapd.ldbm.conf`. Directory Server `slapd` cannot distinguish the difference between `modifytimestamp` and `modifyTimeStamp`.

Incremental dirsync Misses Entries (bug 4164655)

When using a remote directory, if the time of the directory server and the SIMS server are not synchronized, it is possible that incremental dirsync might not pick up all changes in the directory. In such cases, these changes will be reflected in the MTA tables only after the next full dirsync.

External List Member is not Clearly Deleted by the DM Console (bug 4283824)

The last external member of a distribution list is not completely deleted by the DM Console. If you create a distribution list with all external members, then use the DM Console to remove them one by one, everything seems to work properly; no warning messages are displayed. However, upon reverting back to the Administration Console, the last member in the list still remains.

Delegated Manager will not process the delete operation if there are illegal characters at the end of entry. This problem can be resolved by removing the extra <CR>, space, or tab trailing the last member of the distribution list.

imbackup Does Not Backup Shared Folders (bug 4267595)

The `imbackup` utility does not backup Shared folder. For example, take the following command:

```
# imbackup -f /tmp/backup -u /tmp/username
```

where `/tmp/username` is a group members list who used the Shared folder. After invoking `imrestore`, the group members' Inboxs are restored, but the Shared folders are not.

To workaround this problem, use distribution lists instead of Shared folders.

Logging File Systems and SIMS (bug 4247511)

A logging file system is a computer file system that contains its own backup and recovery capability. Before file indexes on disk are updated, the information about the changes are recorded in a log. If a power or other system failure corrupts the indexes as they are being rewritten, the operating system can use the log to repair the indexes when the system is restarted.

SIMS supports two logging file systems, Veritas VxFS and the UFS logging file system. The advantages of using a logging file systems in SIMS are:

- If a message is being written to a folder, and if the system crashes during the write, the folder can be corrupted due to a partial write to disk. Using a logging file system prevents such a partial write.

- When the system crashes or if `imaccessd` is terminated abnormally (for example, by using `kill -9`) you are required to run `imcheck -c` to check and repair message store corruption. This must be done before restarting SIMS. All the SIMS processes must be shut down while `imcheck -c` is running. This can result in hours of downtime. With a logging file system, you are not required to run `imcheck -c` after a system crash or abnormal termination. There is no downtime.

The disadvantages of using a logging file systems in SIMS are:

- Message store performance may be somewhat affected due to file logging overhead.

How to Create an ETRN Queue Channel (bug 4245523)

An `ETRN` Queue Channel can reduce SIMS computational overhead for domains without permanent connections to the mail server. SIMS enqueues messages for disconnected domains and delivers them when the domain client connects to SIMS and sends an SMTP `ETRN <client_domain>` command (see RFC 1985—www.rfc-editor.org). The problem is that by default, SIMS continues attempting delivery of the domain's messages at regular intervals, even though the only time the messages can be delivered is when the domain client is connected. These unsuccessful delivery attempts generate non-delivery messages and waste computational resources.

By creating an ETRN Queue Channel for each domain, messages to that domain are stored in the channel and no delivery attempt is made until the domain client sends an `ETRN` command.

To create an ETRN Queue Channel, you must create a rewrite rule and a channel for each domain which will be using `ETRN`. The rule will cause the mail to be routed to the appropriate channel. The mail will be held in the channel until the client connects and retrieves it.

The rewrite rule should be like the following:

```
domain1.com $E$U%$D@tcp_etrn_dom1-daemon
```

The new channel should have the same settings as the tcp_local channel but also include the `slave` keyword, and possibly change the `notices` keyword:

```
! tcp_etrn_dom1
tcp_etrn_dom1 smtp single_sys subdirs 20 copywarnpost
copysendpost postheadonly immnonurgent noreverse logging
notices 1 2 4 7 blocklimit 10240 charset7 US-ASCII charset8
ISO-8859-1 slave
tcp_etrn_dom1-daemon <fully qualified SIMS host>
```

(note: the `tcp_ertrn_dom1 smpt single_sys...` line above should be all one line, but may appear to be on separate lines because of its length.)

Separate channels are required for each domain because when a client issues the `ETRN` command, the IMTA will attempt to deliver all messages pending in the channel.

For SIMS 4.0, you can have a single ETRN Queue Channel holding the messages for multiple domains. However, the client must issue the ETRN command in the following form:

`ETRN @<client_domain>`

If the client issues this command without the `@`, then SIMS will attempt to deliver all the messages in the channel.

The `slave` keyword prevents the IMTA from attempting to deliver the mail. The messages will be delivered only when the client connects and issues the `ETRN` command. See the SIMS 4.0 Reference Guide page 102 for additional information on the `slave` keyword.

The `notices 1 2 4 7` keyword specifies when warning messages should be returned to senders to let them know that the message is still in a queue waiting to be delivered. Depending on how frequently the remote system will connect to retrieve mail, you may want to increase these values. See the SIMS 4.0 Reference Guide page 108 for additional information on the `notices` keyword.

SMTP AUTH Does Not Support Virtual Domains (bug 4255082)

When using SMTP AUTH, the same mechanism that is used by the Message Access to retrieve a user's credentials in the directory is used when interpreting the identity in the AUTH command issued by the client. In particular, the domain from IP mechanism uses the same set of parameters in `/etc/opt/SUNWmail/ims/ims.cnf`.

SMTP AUTH Supports CRAM-MD5 (bug 4255082)

SMTP AUTH now supports CRAM-MD5 SASL mechanism with the following restrictions:

- When using Sun Directory 3.1, the passwords of the users have to be stored in the directory in clear or using the {sunds} encryption (default).

- When using the Netscape Directory, the passwords have to be stored in clear text in the directory.

To turn on the CRAM-MD5 mechanism, set the following option in `/etc/opt/SUNWmail/imta/option.dat`:

```
SMTPAUTH_USECRAMMD5=1
```

Adding SMTP over TLS to SIMS

This section describes how to add SMTP over TLS to SIMS.

Overview

As defined in RFC 2246, the primary goal of the Transport Layer Security (TLS) protocol is to provide privacy and data integrity between two communicating applications. The TLS protocol itself is based on the SSL 3.0 Protocol Specification as published by Netscape. SIMS 4.0 is compliant with SSL 3.0. There are some differences between TLS 1.0 and SSL 3.0 but TLS 1.0 does incorporate a mechanism by which a TLS implementation can back down to SSL 3.0. In the following, we will talk about TLS and not SSL.

For an overview of SIMS implementation of SSL, see the Chapter 11 of SIMS 4.0 Administrator's Guide, "Secure Sockets Layer (SSL) Support in SIMS."

There are two modes of operation that the SIMS IMTA supports:

- Connecting to a TLS enabled port where TLS negotiation happens immediately once the TCP connection has been established.
- Connecting to a "regular" port and then issuing a STARTTLS (RFC 2487) command to begin TLS negotiation.

The only difference between these two modes is when the TLS negotiation begins. In both cases, once the TLS negotiation is complete, all subsequent data sent across the TCP connection will be secure.

Connecting to a special port number is one way to connect to a TLS enabled server. SMTP has an established port for use with TLS (port 465). When a client connects to this special port (as configured in the Dispatcher configuration file), the IMTA will immediately begin TLS negotiation. Once the negotiation is complete, the connection will be given to the service as usual.

If a STARTTLS command is used, the TCP connection is established on the usual port number (or an alternate port number if configured in the Dispatcher) and given to the service normally. If TLS is available to the client in this SMTP session, the server will advertise STARTTLS as one of its available SMTP extensions. The client

will then issue the STARTTLS command, and the server will acknowledge receipt of the SMTP command and instruct the client to begin TLS negotiation. Again, once the negotiation is complete, the connection continues normally.

If connecting to a special port is largely widespread for IMAP or POP protocols, the STARTTLS command is a better and more flexible choice for SMTP.

Security Layer Configuration

The security layer can be configured on the server or client side.

Server Side Configuration

If you plan to use only the server side of STARTTLS (the server accepts the STARTTLS command but never issues it), or if you want to use the special port, the step by step configuration of TLS/SSL is described in the chapter 11 of SIMS 4.0 Administrator's Guide, "Secure Sockets Layer (SSL) Support in SIMS." If SSL was configured for IMAP/POP, no additional step is needed.

Configuration SMTP Over TLS

This section describes how to configure SMTP over TLS.

Dispatcher Related Configuration For Alternate Port Numbers

By default, the `dispatcher.cnf` file has an SMTP service definition that looks something like:

```
[SERVICE=SMTP]
PORT=25
...
```

To enable TLS for such a dispatcher service, you simply add a `TLS_PORT` option to the configuration for that service. For example, to add TLS support on port 465 for SMTP (the established port for SMTP TLS use), you'd use:

```
[SERVICE=SMTP]
PORT=25
TLS_PORT=465
...
```

Once the dispatcher configuration modifications are complete, you must restart the dispatcher (if it is currently running) or start it (if it is not currently running) so that the new dispatcher configuration with the new port numbers takes effect.

TCP/IP channel configuration for TLS use (STARTTLS)

SIMS supports a number of keywords on the TCP/IP channels to control whether TLS functionality is desired or required. These keywords are summarized in the following table:

TABLE 1 TLS Channel Keywords

Keyword	Usage
`notls`	The combination of `notlsserver` and `notlsclient`; this is the default
`maytls`	The combination of `maytlsserver` and `maytlsclient`
`musttls`	The combination of `musttlsserver` and `musttlsclient`
`notlsserver`	Do not offer the STARTTLS extension and do not accept a STARTTLS command from a remote client
`maytlsserver`	Offer and accept STARTTLS (if not already TLS enabled)
`musttlsserver`	Offer and require STARTTLS (if not already TLS enabled); if TLS has not been negotiated, refuse to accept any mail during this session with a "530" error
`notlsclient`	Do not attempt to use STARTTLS even if offered by a remote SMTP server
`maytlsclient`	If STARTTLS is offered by a remote SMTP server, attempt to use TLS
`musttlsclient`	Use STARTTLS if offered by a remote SMTP server, but if not available, this message delivery will be aborted
`tlsswitchchannel` *`channelname`*	If TLS is used, switch to the channel specified as the channelname parameter to this keyword

Enabling (or requiring) the use of TLS may be of interest with dedicated channels intended for communicating sensitive information with companion systems that also support TLS.

Enabling the use of TLS for the SMTP server may also be of particular interest when SMTP SASL use has been enabled. Since with SMTP SASL use, a remote client will be sending a password over the network, then, especially if the PLAIN authentication mechanism is used (password sent "in the clear"), it may be particularly desirable to use TLS so that the entire transaction, including the password, is encrypted.

Use of the `tlsswitchchannel` keyword may be of interest for logging purposes, so that log entries show the message as coming in via a special channel. Use of the `tlsswitchchannel` keyword may also be of interest if it is desired to route messages submitted using TLS differently (using source channel specific rewrite rules) than messages submitted without TLS.

About the Switchchannel Keywords

In the following examples, we make an extensive use of the `switchchannel`/`tlsswitchchannel`/`saslswitchchannel` keywords. Those keywords allow you to switch the source channel of a connection. Switching the source channel means that all the messages submitted by the user will be seen (in terms of logging or access control) as coming from the new source channel. Another effect of switching the source channel is that all the keywords associated with the old source channel are forgotten and the ones associated with the new channel apply. For example, if the `tcp_local` channel definition contains the `musttlsserver` keyword but the user is switched to `tcp_intranet`, which doesn't contain this keyword, the user won't have to use TLS.

If the switchchannel is configured in the tcp_local definition (see SIMS 4.0 Reference Guide, "Selecting an Alternate Channel for Incoming Mail" page 118), it is applied first, at the beginning of the connection. For the other switchchannel keywords, they are taken into account in the order the corresponding commands are issued.

Sample TLS Configurations

The following three examples describe how to configure TLS.

Example 1

A site that has a submission SMTP server reserved for its own subscribers and has the following policy:

- Any subscriber connecting from outside the intranet must use TLS.
- Any subscriber connecting from inside the intranet may use TLS.

The system can be configured as follows (only the main keywords are shown):

in `imta.cnf`:

```
tcp_local smtp switchchannel musttlsserver tlsswitchchannel tcp_tls
tcp-daemon

tcp_intranet smtp mx single_sys maytlsserver tlsswitchchannel tcp_tls
tcp_intranet-daemon

tcp_tls smtp mx single_sys musttlsserver
tcp_tls-daemon
```

A subscriber trying to connect from tcp_local must issue the STARTTLS command in order to be able to send a message (`musttlsserver` keyword). If the command succeeds, the user is switched to the tcp_tls channel (`tlsswitchchannel tcp_tls` keyword), every message submitted by this user will be seen as coming from the tcp_tls channel (for logging as well as access mapping purpose).

A user trying to connect from the site's intranet will be first switched from tcp_local to tcp_intranet. Consequently, STARTTLS is offered to him, but he doesn't have to use it (`maytlsserver` keyword). If he issues the STARTTLS command, he will be switched to tcp_tls.

Of course a publicly referenced SMTP server (MX recorded) shouldn't use this kind of configuration.

Example 2

A site that generally blocks SMTP relaying through their SMTP server, but wishes to allow such SMTP relaying for specific users who will authenticate themselves using SASL (SMTP AUTH), might use channel definitions similar to these given below. This type of configuration is particularly appropriate for sites wanting to allow roaming users to keep relaying mail through their domain's mail server, while preventing other users to do the same.

In `imta.cnf`:

```
tcp_local smtp mx single_sys maysaslserver saslswitchchannel tcp_auth
tcp-daemon

tcp_auth smtp mx single_sys mustsaslserver
tcp-auth-daemon
```

with an ORIG_SEND_ACCESS mapping table (`/etc/opt/SUNWmail/imta/mappings`) like this:

```
ORIG_SEND_ACCESS

tcp_local|*|tcp_local|* $NRelaying$ not$ permitted
```

For details about using SASL, see SMTP AUTH Configuration on page 140 of the SIMS 4.0 Administrator's Guide.

The problem with this configuration is that clients will use the PLAIN mechanism to authenticate. The PLAIN mechanism implies that user passwords are sent in clear text. Passwords should never be sent in clear text in an untrusted environment unless over TLS.

The same configuration with TLS would look like this:

In `imta.cnf`:

```
tcp_local smtp mx maytlsserver tlsswitchchannel tcp_tls
tcp-daemon

tcp_tls smtp mx musttlsserver maysaslserver saslswitchchannel tcp_auth
tcp_tls-daemon

tcp_auth smtp mx mustsaslserver musttlsserver
tcp-auth-daemon
```

with an `ORIG_SEND_ACCESS` mapping table (`/etc/opt/SUNWmail/imta/mappings`) like this:

```
ORIG_SEND_ACCESS

tcp_auth|*|*|* $Y
tcp_*|*|tcp_local|* $NRelaying$ not$ permitted
```

A client connecting from the tcp_local channel issues the `EHLO` command. The server offers `STARTTLS` (`maytlsserver` keyword in tcp_local definition) but not `AUTH` as an extension. The client issues the `STARTTLS` command and is switched to the `tcp_tls` channel (`tlsswitchchannel tcp_tls` keyword in tcp_local definition).

Then, it issues a new `EHLO` command. At this point, since the `maysaslserver` keyword is configured for the `tcp_tls` channel, the server offers `AUTH` in the available extensions. If the client authenticates successfully, it is switched to `tcp_auth` (`saslswitchchannel tcp_auth` keyword in `tcp_tls` definition) and the `ORIG_SEND_ACCESS` rules apply from this channel.

Example 3

Three companies (a.com, b.com, and c.com) want to exchange secure information over the Internet. They want to use TLS when sending messages to each other but not when talking to any other domain. A sample configuration for a.com could be:

In `imta.cnf`:

```
! Rules to select local users
a.com    $E$U%$D@myhost.a.com
! My buddy rules
b.com $E$U%$D@tcp_tls-daemon
c.com $E$U%$D@tcp_tls-daemon
! Rules for top level internet domains
</etc/opt/SUNWmail/imta//internet.rules
. $E$U%$H@tcp-daemon
...

l noswitchchannel...
myhost.a.com

tcp_local switchchannel smtp mx maytlsserver tlsswitchchannel tcp_tls
tcp-daemon

tcp_tls smtp mx musttlsserver musttlsclient
tcp_tls-daemon
```

A message for `user@toto.com` matches a rule in `internet.rules` (`.com $U%$H$D@tcp-daemon`) and goes into the `tcp_local` channel from which it will be sent without using TLS.

A message for `user@b.com` matches the (`b.com $E$U%$D@tcp_tls-daemon`) rule and consequently is routed to the `tcp_tls` channel. The `tcp_tls` channel definition contains the **`musttlsclient`** keyword. When the message is finally sent to b.com mailserver, the `tcp_smtp_client` must use STARTTLS.

New Option for imquotacheck

A new command line option has been added to the `imquotacheck` utility. The `-m` option allows the administrator to customize the quota warning message. The syntax for the `-m` option is:

```
-m <msg-file>
```

where the <msg-file> variable represents the file name of the quota warning message. The <msg-file> must contain a valid message header and a message body. Every line must be terminated with a CRLF. Required header fields must be present. The following macros can be used inside the message body (`imquotacheck` replaces the macros with the user's information before it delivers the message):

$DATE	Current date in RFC 822 format
$STOREOWNER	Message store owner
$USERID	User's message store userid
$QUOTA	User's quota limit
$PERCENT	Percent used
$USAGE	Current disk usage for user

The following is an example <msg-file> (`quota.msg`):

```
Date: $DATE
From: $STOREOWNER
TO: $USERID
Subject: WARNING: LOW QUOTA
Mime-Version: 1.0
Content-Type: TEXT/PLAIN; CHARSET=US-ASCII

Dear $USERID,

Your total mailbox size has exceeded $PERCENT% of the assigned
quota:
Mailbox size = $USAGE
Quota = $QUOTA

Thanks for using our email server.

Your email administrator
```

Using the above `quota.msg` file, the following command can be executed:

```
# imquotacheck -u joe -m quota.msg
```

The following is the example output:

```
Date: Mon, 23 Aug 1999 14:58:07 -0700 (PDT)
From: inetmail
TO: joe
Subject: WARNING: LOW QUOTA
Mime-Version: 1.0
Content-Type: TEXT/PLAIN; CHARSET=US-ASCII

Dear joe,

Your total mailbox size has exceeded 100% of the assigned quota:
Mailbox size = 5039
Quota = 5000

Thanks for using our email server.

Your email administrator
```

Documentation Updates

This section covers additions and corrections to SIMS 4.0 documentation for the SIMS 4.0 patch 6 release.

Information about LDAPHOST_TIMEFLAG is not Documented

The folowing information was not included in the *SIMS 4.0 Reference Manual*. LDAPHOST_TIMEFLAG is not an implemented option in SIMS 4.0. Use IMTA_INCREMENTAL_TIMESKEW instead. Bug 4299505 states: Incremental dirsync was missing a few entries because of the time skew between the directory and mail server. This problem has been fixed, but incremental dirsync must set the time stamp only after taking into account the time skew between directory server and the mail server. An option, IMTA_INCREMENTAL_TIMESKEW, can be set in dirsync.opt. This value should be set to the number of seconds that reflects the time lag between the machine that runs the MTA and the machine that runs the directory service.

Description of Headerset8 is Missing in SIMS 4.0 Documentation (bug 4299541)

The following information was not included in the *SIMS 4.0 Reference Manual*. The `headerset8` channel keyword is designed to allow non-ASCII message headers to be delivered without RFC2047 encoding. This is popularly referred to as “Just Send 8” message headers. Setting `headerset8` on the destination channel will cause SIMS to remove the MIME encoding on the Subject: lines and on other unstructured text header lines, including headers containing addresses.

When “Just Send 8” messages are submitted to the IMTA, the source channel normalizes the headers into RFC2047 format. The definition of the charset tag depends on the configuration of the source channel. If the source channel is unmarked, the charset label is set to `UNKNOWN`. In theory, by setting `headerset8 UNKNOWN` on the destination channel, the headers would be restored to their original un-encoded form.

Normally headerset-originating decoding or any other kind of header decoding is not performed from the `UNKNOWN` character set. A code change in SIMS 4.0 consists in allowing `UNKNOWN` charset decoding when configured by the user through a headerset keyword. A plausible but untested workaround would be to use the `charset8` channel keyword on the source channel to set a fictitious character set:

```
charset8 FAKE-CHARSET
```

then set the `headerset8` on the destination to the same value. To prevent the fictitious name from appearing in the Content-Type field, it would be necessary to set up a charset conversion relabelling operation.

Note – Using the `headerset8` channel keyword produces non-RFC-compliant data and may cause serious problems for the receiving SMTP servers. Sun highly discourages customers from implementing this keyword.

Correction to Documentation on `dns_verify` Arguments (bug 4328992)

In the SIMS 4.0 Administrator’s Guide, DNS-based Email Access Control is described on pages 128-132. The following corrections to the text should be noted:

Page 128, the 4th paragraph should read:

Given a domain, the `dns_verify` program queries the DNS to check if either

an A record or an MX record exists for the domain. Then depending on the way you configure it, the host can be accepted or rejected.

Page 129, dns_verify Arguments, should read:

The arguments to `dns_verify` has four parts delimited by '|' as shown below (or delimited by another character if | is part of the template, see more examples later using + as the delimiter):

```
hostname|return-if-good|return-if-bad|return-if-indetermined
```

The `return-if-good` and `return-if-bad` and `return-if-indetermined` strings are templates for the return string. If hostname has a valid DNS entry, the `return-if-good` template is used to generate a return string; if the DNS server returns a authoritative "host not found" error, then the `return-if-bad` template is used to generate a return string; `return-of-undetermined` template is used if the dns lookup gives a `SERVFAIL` or other "indeterminant" error, meaning we can NOT determine if the host name exists in the DNS or not.

If the `return-if-undetermined` template is not given, you only used the three parts of the argument:

```
hostname|return-if-good|return-of-bad
```

then in the case of undetermined returns from the DNS lookup, `return-if-good` is used.

Example

```
$2|$$Y|$$NInvalid$ host:$ $$2$ -$ %e|$$NCannot$ look$ up$ host:$ $$2$
-$ %e
```

One page 128, in the example boxes, a blank line is required after the `ORIG_SEND_ACCESS` line, and the rest of the lines need to be indented.

On pages 130, in the example box, a blank line is required after the `PORT_ACCESS` line, and the rest of the lines need to be indented.

Additional Information about `imta.cnf` Channel Blocks (bug 4335729)

It is possible to configure the MTA to advertise a name other than the default hostname in the SMTP banner and the HELO and EHLO messages. In the channel block, the line which the SIMS documentation calls the “routing system identifier” can do this:

```
! tcp_local
tcp_local smtp single_sys ...
tcp-daemon zortch proxycache.labnet.east.sun.com
```

Hostname “zortch” will now be used in the SMTP banner and the HELO and EHLO messages.

`imquotacheck` Man Page is Missing -f Option (bug 4334742)

The `imquotacheck` command sends an email warning to users who are approaching their mail store quota. This command can be put in a cron file to provide a daily check on mail store users. You can configure the desired warning message using the -f flag for `imquotacheck`.

imsauth(5)Man Page Change (bug 4320666)

A new member, internal_id, was added to auth_user_info structure. It is to be ignored by the library author. The complete list of members are:

```
struct auth_user_info{

              char *uid;
              char *passwd;
              char *mailhost;
              char **admin_attr;
              char *userstatus;
              char **user_authsrvc;
              char *user_domain;
              char *domain_info;
              char *mstore;
              char *inbox;
              char *u_attr;
              char *homedir;
              char *internal_id; <<======= new member
```

In addition, make note of the following correction:

The last five fields of the structure should be ignored by the library author. They are all related to /var/mail mailboxes, and this is not supported for the imsauth plug-in interface. The last one, internal_id, is for Sun's internal use only.

SEND_ACCESS Mapping Missing from SIMS 4.0 Doc (bug 4265689)

The following information was not included in the *SIMS 4.0 Reference Manual.* The mappings file, /etc/opt SUNWmail/imta/mappings includes a SEND_ACCESS table by default. The SEND_ACCESS or ORIG_SEND_ACCESS mapping table may be used to control who may or may not send mail, receive mail, or both. If a SEND_ACCESS or ORIG_SEND_ACCESS mapping table exists, then for each recipient of every message passing through the IMTA, the IMTA will probe the table with a probe string of the form (note the use of the vertical bar character):

src-channel|*from-address*|*dst-channel*|*to-address*

src-channel is the channel originating the message (i.e.. queueing the message): *from-address* is the address of the message's originator; *dst-channel* is the channel to which the message will be queued; and *to-address* is the address to which the message is addressed. Use of an asterisk in any of these four fields causes the fields to match any channel or address as appropriate.

Now, if the probe string matches a pattern (i.e., the left hand side of an entry in the table), then the resulting output of the mapping is checked. If the output contains the metacharacters $N, then the enqueue to that particular address is rejected. In the case of a rejection, optional rejection text may be supplied in the mapping output. This string will be included in the rejection message. If no string is output (other than the `$N` metacharacter), then a default message will be used.

Suppose that local users in the domain acme.com, with the exception of the postmaster, are not allowed to send mail to the Internet, but can receive mail from there. Then the `SEND_ACCESS` mapping table shown below is one possible way to enforce this restriction. In that example, the local host name is assumed to be acme.com. In the channel name `tcp_*`, wild cards are used so as to match any possible TCP/IP channel name. in the rejection message, dollar signs are used to quote spaces in the message. Without the dollar signs, the rejection would be ended prematurely and only read "Internet" instead of "Internet postings are not permitted."

SEND_ACCESS

```
 *|postmaster@acme.com|*|*   $YN00kIe
 *|*|*postmaster@acme.com    $Y
 1|*@acme.com|*tcp_*|*       $NInternet$ postings$ are$ not$
permitted
```

Web Access Updates

This section describes the updates for the Web Access client.

Autoresponder for Vacation Mail Creates Non-Compliant MIME Messages (Bug 4281880)

When a user's vacation mail is set and enabled, sending mail to this user creates a mail message issued by the autoresponder which does not contain the MIME-version header for the message, making it a non-compliant MIME message.

Login Separator Inconsistency (Bug 4250347)

In the case where Web Access and SIMS are installed on the same system at the same time, the SIMS installation mechanism configures the separator for both Web Access and SIMS. The only way the separator can be different in this case is if the administrator intentionally changes it after installation.

In the case where SIMS and Web Access are installed on separate systems or at different times, the SIMS installation GUI is used to install both components and configure the separator for both. The administrator can specify different separators in this case.

If the login separators are different for SIMS and Web Access, a user may be able to successfully log in to Web Access, but cannot read mail.

Enabling Instances via SWS2.1 (Bug 4250317 and 4250498)

In order to start and stop the Web Access server using the `htserver` command, the instances must be enabled via the SWS2.1 (Sun Web Server):

```
# htserver enable sws_server
# htserver enable admin
# htserver enable WebAccess
```

You may now stop and start Web Access:

```
# htserver stop WebAccess
```

```
# htserver start WebAccess
```

Starting and Stopping Web Access

You can start and stop the Web Access server using the `webaccess start` and `webaccess stop` commands (using SIMS 4.0).

To stop the Web Access server:

```
# /etc/init.d/webaccess stop
```

To start the Web Access server:

```
# /etc/init.d/webaccess start
```