



we protect your digital worlds

ESET Mail Security

*Installation Manual
and User's documentation*

Table of contents

1. Introduction	3
2. Terminology and abbreviations	5
3. Installation	9
4. Product's Roadmap	11
5. Integration with E-mail Messaging System	15
5.1. Bi-directional e-mail messages scanning in MTA	17
5.2. Scanning of inbound e-mail messages	17
5.3. Scanning of outbound e-mail messages	18
5.4. Scanning of e-mail messages being downloaded from POP3/IMAP server	18
5.5. Alternative methods of content filtering	18
5.5.1. Scanning e-mail messages using AMaViS	18
5.5.1.1. amavis	19
5.5.1.2. amavisd	20
5.5.1.3. amavisd-new	20
5.5.2. Scanning e-mail messages in KerioMailServer	20
6. Important ESET Mail Security mechanisms	21
6.1. Handle Object Policy	22
6.2. User Specific Configuration	23
6.3. Black-list and white-list	24
6.4. Anti-Spam Control	24
6.5. Samples Submission System	25
6.6. World WideWeb Interface	25
7. ESET Mail Security system update	27
7.1. ESETS update utility	28
7.2. ESETS update process description	28
8. Tips and Tricks	31
8.1. ESETS and TLS support in MTA	32
9. Let us know	33
A. ESETS setup process description	35
A.1. Setting ESETS for MTA Postfix	36
A.1.1. Inbound e-mail messages scanning	36
A.1.2. Bi-directional e-mail messages scanning	36
A.2. Setting ESETS for MTA Sendmail	37
A.2.1. Inbound e-mail messages scanning	37
A.2.2. Bi-directional e-mail messages scanning	38
A.3. Setting ESETS for MTA Qmail	38
A.3.1. Inbound e-mail messages scanning	38
A.3.2. Bi-directional e-mail messages scanning	39
A.4. Setting ESETS for MTA Exim version 3	39
A.4.1. Inbound e-mail messages scanning	39
A.4.2. Bi-directional e-mail messages scanning	40
A.5. Setting ESETS for MTA Exim version 4	40
A.5.1. Inbound e-mail messages scanning	40
A.5.2. Bi-directional e-mail messages scanning	41
A.6. Setting ESETS for outbound e-mail messages scanning	41
A.7. Setting ESETS for scanning of POP3 communication	41
A.8. Setting ESETS for scanning of IMAP communication	42
B. PHP License	43

ESET Mail Security
Copyright © 2007 ESET, spol. s r.o.

ESET Mail Security was developed by
ESET, spol. s r.o. For more information
visit www.nod32.com.hk

All rights reserved. No part of this
documentation may be reproduced,
stored in a retrieval system or
transmitted in any form or by any
means, electronic, mechanical,
photocopying, recording, scanning,
or otherwise without a permission in
writing from the author. ESET, spol. s
r.o. reserves the right to change any
of the described application software
without prior notice. This product
includes PHP software, freely available
from <http://www.php.net/software/>.
ESET Mail Security was developed in
co-operation with ProWeb Consulting.
For more information visit www.pwc.sk.

REV.20071112-003



Chapter 1:

Introduction



Dear user, you have acquired ESET Mail Security - probably the best security system running under the Linux/BSD OS. As you will soon find out, the system using the state-of-the-art ESET scanning engine, has unsurpassed scanning speed and detection rate, combined with a very small footprint that makes it the ideal choice for any Linux/BSD OS server.

In the rest of this chapter we review a key features of the system.

- The ESET anti-virus scanning engine algorithms provide the highest detection rate and the fastest scanning times.
- The ESET Mail Security is developed to run on the single-processor as well as on the multi-processor units.
- It includes unique advanced heuristics for Win32 worms and back-doors.
- Inbuilt archivers unpack archived objects without the need for any external programs.
- In order to increase speed and efficiency of the system, its architecture is based on the running daemon (resident program) where all the scanning requests are sent to.
- The system supports selective configuration specific for user or client/server identification.
- Six logging levels can be configured to get information about system activity and infiltrations.
- The ESET Mail Security installation does not require external libraries or programs except for LIBC.
- The system can be configured to notify any person in case of detected infiltration.
- The system contains anti-spam control mechanism.
- Information about infiltration can be configured to be written into an e-mail header, footer and subject.

To run efficiently, ESET Mail Security requires just 16MB of hard-disk space and 32MB of RAM. It works smoothly under the 2.2.x, 2.4.x and 2.6.x Linux OS kernel versions and also under 5.x, 6.x FreeBSD OS kernel versions.

From lower-powered, small office servers to enterprise-class ISP servers with thousands of users, the system delivers the performance and scalability you expect from a UNIX based solution and the unequalled security of ESET products.



Chapter 2:

Terminology and abbreviations



In the following text we review terms and abbreviations used in this documentation. Note that in this documentation (PDF format only) a boldface font is reserved for product components names and in this chapter also for newly defined terms and abbreviations. Note also that terms and abbreviations defined in this chapter are emphasized later in this documentation (PDF format only).

ESETS

ESET Security is a common acronym for all security products developed by ESET, spol. s r.o. for Linux OS (resp. for BSD OS). It is also the name (or its part) of the software package containing the products.

RSR

Abbreviation of 'RedHat/Novell(SuSE) Ready'. Note that we support also so called RedHat Ready and Novell(SuSE) Ready variation of the product. The difference from the "standard" Linux version is that the RSR package meets criteria defined by FHS (File-system Hierarchy Standard defined as a part of Linux Standard Base) document required by the RedHat Ready and Novell(SuSE) Ready certificate. This means for instance that the RSR package is installed as an add-on application, i.e. the primary installation directory is `/opt/eset/esets`.

ESETS daemon

Main *ESETS* system control and scanning daemon `esets_daemon`.

ESETS base directory

The directory where *ESETS* loadable modules containing for instance virus signatures database are stored. Further in this documentation we use abbreviation `@BASEDIR@` for the directory. The directory location is as follows:

```
Linux: /var/lib/esets
Linux RSR: /var/opt/eset/esets/lib
BSD: /var/lib/esets
```

ESETS configuration directory

A directory where all files related with the ESET Mail Security configuration are stored. Further in this documentation we use abbreviation `@ETCDIR@` for the directory. The directory location is as follows:

```
Linux: /etc/esets
Linux RSR: /etc/opt/eset/esets
BSD: /usr/local/etc/esets
```

ESETS configuration file

Main ESET Mail Security configuration file. The absolute path of the file is as follows:

```
@ETCDIR@/esets.cfg
```

ESETS binary files directory

The directory where the relevant ESET Mail Security binary files are stored. Further in this

documentation we use abbreviation **@BINDIR@** for the directory. The directory location is as follows:

```
Linux: /usr/bin
Linux RSR: /opt/eset/esets/bin
BSD: /usr/local/bin
```

ESETS system binary files directory

The directory where the relevant ESET Mail Security system binary files are stored. Further in this documentation we use abbreviation **@SBINDIR@** for the directory. The directory location is as follows:

```
Linux: /usr/sbin
Linux RSR: /opt/eset/esets/sbin
BSD: /usr/local/sbin
```

ESETS object files directory

The directory where the relevant ESET Mail Security object files and libraries are stored. Further in this documentation we use abbreviation **@LIBDIR@** for the directory. The directory location is as follows:

```
Linux: /usr/lib/esets
Linux RSR: /opt/eset/esets/lib
BSD: /usr/local/lib/esets
```





Chapter 3:

Installation



This product is distributed as a binary file:

```
eSETS.i386.ext.bin
```

where 'ext' is a Linux/BSD OS distribution dependent suffix, i.e. 'deb' for Debian, 'rpm' for RedHat and SuSE, 'tgz' for other Linux OS distributions, 'fbs5.tgz' for FreeBSD 5.xx and 'fbs6.tgz' for FreeBSD 6.xx distributions.

Note that the Linux *RSR* binary file format is:

```
eSETS-rsr.i386.rpm.bin
```

In order to install or update the product, use statement:

```
sh ./eSETS.i386.ext.bin
```

resp. for Linux *RSR* variation of the product, use statement:

```
sh ./eSETS-rsr.i386.rpm.bin
```

As a result the product's User License Acceptance Agreement is shown. Once you have confirmed the Acceptance Agreement, the installation package is placed into the current working directory and relevant information regarding the package's installation, un-installation or update is printed into terminal.

Once the package is installed and the main *ESETS* service is running, in Linux OS you can check its operation by using command:

```
ps -C eSETS_daemon
```

In case of BSD OS you can use a similar command:

```
ps -ax eSETS_daemon | grep eSETS_daemon
```

You will see the following (or similar) message on return:

PID	TTY	TIME	CMD
2226	?	00:00:00	eSETS_daemon
2229	?	00:00:00	eSETS_daemon

where at least two *ESETS daemon* processes running in the background have to be present. One of the processes is so-called process and threads manager of the system. The other serves as *ESETS* scanning process.



Chapter 4:

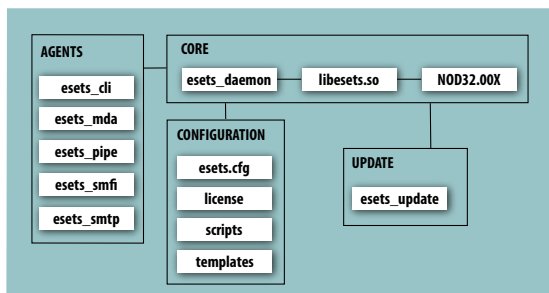
Product's Roadmap



Once the product package has been successfully installed, it is time to become familiar with its content.

The structure of ESET Mail Security is shown in the figure 4-1. The system is composed of the following components.

Figure 4-1. Structure of ESET Mail Security.



CORE

Core of ESET Mail Security consists of *ESETS daemon* `esets_daemon`. The daemon uses *ESETS API* library `libesets.so` and *ESETS* loading modules `nod32.00X` to provide base system tasks: scanning, maintenance of the agent daemon processes, maintenance of the samples submission system, logging, notification, etc.. Please refer to `esets_daemon(8)` manual page for details.

AGENTS

The purpose of *ESETS* agent modules is to integrate *ESETS* with the Linux/BSD Server environment. Please note a special chapter in this document devoted to the topic.

UPDATE

The update utility is a particular fraction of the system. It is developed to update *ESETS* loading modules containing for instance virus signatures database, archives support, advanced heuristics support etc. Please note a special chapter in this document devoted to the topic.

CONFIGURATION

Proper configuration is the most important condition for the system operation. Therefore we describe all the related components in the rest of this chapter. We also strongly recommend to read `esets.cfg(5)` manual page, an essential information source regarding *ESETS* configuration.

After the product is successfully installed, all its configuration components are stored in *ESETS configuration directory*. The directory consists of the following files.

@ETCDIR@/esets.cfg

This is the most important configuration file as it maintains the major part of the product functionality. After exploring the file you can see that it is built from various parameters distributed within sections. Note the section names always enclosed in square brackets. In

the *ESETS configuration file* there is always one global and several so-called agent sections. Parameters in global section are used to define configuration options of *ESETS daemon* as well as default values of *ESETS* scanning engine configuration options. Parameters in agent sections are used to define configuration options of so-called agents, i.e. modules used to intercept various data flow types in the computer and/or its neighborhood and prepare this data for scanning. Note that besides the number of parameters used for the system configuration, there is also a number of rules determining organization of the file. To become familiar with this knowledge, please refer to *esets.cfg(5)*, *esets_daemon(8)* manual page and also to manual pages related to relevant agents.

@ETCDIR@/certs

This directory is used to store the certificates used by *ESETS WWW* Interface for authentication (see *esets_wwwi(8)* for details).

@ETCDIR@/license

This directory is used to store the product(s) license key(s) you have acquired from your vendor. Note that the *ESETS daemon* will always check only this directory to evaluate license key validity unless it is redefined by *ESETS configuration file* parameter 'lic_dir'.

@ETCDIR@/scripts/license_warning_script

This script, if enabled by *ESETS configuration file* parameter 'license_warn_enabled', is executed since 30 days (once per day) before product license expiration. It is used to send e-mail notification about the expiration status to system administrator.

@ETCDIR@/scripts/daemon_notification_script

This script, if enabled by *ESETS configuration file* parameter 'exec_script', is executed in case the infiltration has been detected by the anti-virus system. It is used to send e-mail notification about the event to system administrator.

@ETCDIR@/anti-spam

This directory contains configuration file used to fine tune the anti-spam engine operation.

@ETCDIR@/templates/mail_sig_*.html.example

These files are html templates used to define text of messages inserted as a footnote into the scanned e-mails. To enable these html templates the 'example' suffix must be removed from all of the template file names. Note also that the appearance of the e-mail messages footnotes is maintained by *ESETS configuration file* parameter 'write_to_footnote'. The meaning of individual template files is as follows.

The following footnote templates are used in e-mails found as infected:

e-mail header	From:
.	To:

e-mail body	text of e-mail body
.	content of lms_sig_header_infected.html
.	list of infiltrations found by the scanner
.	content of lms_sig_footer_infected.html

The following footnote templates are used in e-mails found as clean:

e-mail header	From:
.	To:

e-mail body	text of e-mail body
.	content of lms_sig_header_clean.html
.	list of objects scanned by the scanner
.	content of lms_sig_footer_clean.html

The following footnote templates are used in e-mails that could not be scanned:

e-mail header	From:
.	To:

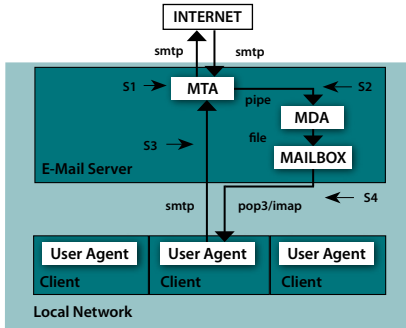
e-mail body	text of e-mail body
.	content of lms_sig_header_not_scanned.html
.	list of object scanned by the scanner
.	content of lms_sig_footer_not_scanned.html

Chapter 5:

Integration with E-mail Messaging System

This chapter describes integration of the ESET Mail Security with the variety of known e-mail messaging systems. Knowledge of e-mail messaging system basic principles (figure 5-1) is of paramount importance for understanding of ESETS operation.

Figure 5-1. Scheme of UNIX OS e-mail messaging system.



MTA - Mail Transport Agent

A program (for instance sendmail, postfix, qmail, exim, etc.) providing e-mail messages transfer among local and remote domains.

MDA - Mail Delivery Agent

A program (for instance maildrop, procmail, deliver, local.mail, etc.) providing delivery of locally addressed e-mail messages into particular mailboxes.

MUA - Mail User Agent

A program (for instance MS Outlook, Mozilla Mail, Eudora, etc.) providing access and management of e-mail messages, i.e. reading, composing, printing etc., stored in mailboxes.

MAILBOX

A file or a file structure on a disk serving as the storage space for e-mail messages. Note that there are several formats of MAILBOX in Linux/BSD OS: an old fashioned format where e-mails for each user are stored sequentially in one user appropriate file located in directory `'/var/spool/mail'`; MBOX (a bit newer but still an old format) with e-mails stored sequentially in one file located within user home directory; MAILDIR with e-mails stored in a separate files within a hierarchical directory structure.

The e-mail server receives data communication typically using SMTP - Simple Mail Transfer Protocol communication. The received message is transferred by MTA either to another remote e-mail messaging system or it is delivered using local MDA into particular MAILBOX (we assume each local network user owns a MAILBOX located at the server disk). Note that it is responsibility of the user's local MUA to provide download and correct interpretation of the message at the user's computer. When retrieving data from MAILBOX the MUA uses typically POP3 - Post Office

Protocol or IMAP - Internet Message Access Protocol to communicate with the MTA. To send data to the Internet the SMTP protocol communication is used.

The *ESETS* operating principle is based on data communication interception and scanning at the various phases of its transfer. The interception locations are marked in the figure 4-1 by symbols S1, S2, S3 and S4.

S1

Bi-directional e-mail messages scanning, i.e. content filtering in MTA.

S2

Scanning of inbound e-mail messages, i.e. messages with the target address corresponding to the destination located inside the local domain.

S3

Scanning of outbound e-mail messages, i.e. messages bound to some remote Internet domain via its target address.

S4

Scanning of e-mail messages being downloaded from POP3/IMAP server.

The rest of this chapter reviews methods of integration of *ESETS* with variety of supported messaging systems.

5.1. Bi-directional e-mail messages scanning in MTA

The advantage of bi-directional e-mail messages scanning mode is that it allows one to scan e-mails inbound as well as outbound in the same implementation algorithm. On the other hand the bi-directional (content filter) method is MTA dependent. The ESET comes with four content filters built for most common MTA, i.e. MTA Sendmail, Postfix, Exim and QMail.

In order to configure ESET Mail Security for bi-directional e-mail messages scanning you have to be sure that your MTA is properly configured and running. Then run this script:

```
esets_setup
```

Select MTA and contentfilter install options. Used *ESETS* module is also displayed.

Note that the installer backups all modified configuration files and can display all commands it will execute after your approval. Use it for uninstall, too. The detailed steps for all possible scenarios are described in the appendix A of this documentation.

5.2. Scanning of inbound e-mail messages

Scanning of the inbound e-mail messages is performed during the messages transfer between MTA and MDA. The incoming e-mail is intercepted by **esets_mda** module, scanned by *ESETS daemon* and delivered to MAILBOX using original MDA. As shown in the figure, the

virus scanning can be enabled by proper configuration setting of MTA and **esets_mda** module. Note that the ESET Mail Security supports most common MTA, i.e. MTA Sendmail, Postfix, Exim and QMail. *ESETS* supports any MDA. In particular the following MDAs were tested: procmail, maildrop, deliver and local.mail.

In order to configure ESET Mail Security for inbound e-mail messages scanning you have to be sure that your MTA is properly configured using original MDA and running. Then run this script:

```
esets_setup
```

Select MDA and inbound install options. Used *ESETS* module is also displayed.

Note that the installer backups all modified configuration files and can display all commands it will execute after your approval. Use it for uninstall, too. The detailed steps for all possible scenarios are described in the appendix A of this documentation.

5.3. Scanning of outbound e-mail messages

Scanning of the outbound e-mail messages is performed during transfer of e-mail messages between the local MUA and the MTA.

In order to configure ESET Mail Security for outbound e-mail messages scanning run this script:

```
esets_setup
```

Select SMTP install option. It will setup **esets_smtp** module to listen on predefined port and redirect applicable IP packets. Check added firewall rule and move it or change according your needs.

Note that the installer backups all modified configuration files and can display all commands it will execute after your approval. Use it for uninstall, too. The detailed steps for all possible scenarios are described in the appendix A of this documentation.

5.4. Scanning of e-mail messages being downloaded from POP3/IMAP server

In order to configure ESET Mail Security for scanning of e-mail messages downloaded from POP3 (resp. IMAP) server run this script:

```
esets_setup
```

Select POP3 or IMAP install option. It will setup displayed *ESETS* module to listen on predefined port and redirect applicable IP packets. Check added firewall rule and move it or change according your needs.

Note that the installer backups all modified configuration files and can display all commands it will execute after your approval. Use it for uninstall, too. The detailed steps for all possible scenarios are described in the appendix A of this documentation.

5.5. Alternative methods of content filtering

5.5.1. Scanning e-mail messages using AMaViS

AMaViS - A Mail Virus Scanner is a tool that interfaces your MTA and several anti-virus scanners. It supports various MTAs and comes in three branches: **amavis**, **amavisd** and **amavisd-new**. Amavis cooperates with ESET Mail Security by using **esets_cli**. Yet before we go into detailed explanation of the Amavis configurations, we would like to discuss the impact of the method on the ESET Mail Security functionality.

First, note that Amavis does not allow modification of the scanned e-mail messages. So no infected e-mail attachments can be cleaned nor deleted by *ESETS*. Second consequence is that no *ESETS* footnote with log and status dependent header fields will be written into the e-mail. Next, amavis doesn't provide mail sender/recipient, so no user specific configurations can be used, too. Advanced mail handling (accept, defer, discard, reject) is also limited for **esets_cli**. Lastly, it scans files and thus cannot use *ESETS* anti-spam engine.

Taking into account these drawbacks, this configuration is usable if only the above discussed features of the product are not necessary for the user.

5.5.1.1. amavis

Configuration of Amavis is performed during the Amavis installation. After unpacking the source `amavis-0.x.y.tgz`, create the file `amavis/av/esets_cli` with this contents:

```
#
# ESET Software ESETS Command Line Interface
#
if ($esets_cli) {
  do_log(2, "Using $esets_cli");
  chop($output = `$esets_cli --subdir $TEMPDIR/parts`);
  $errval = retcode($?);
  do_log(2, $output);
}
if ($errval == 0) {
  $scanner_errors = 0;
}
elseif ($errval == 1 || $errval == 2 || $errval == 3) {
  $scanner_errors = 0;
  @virusname = ($output =~ /virus="([^\"]+)"/g);
  do_virus();
}
else {
  do_log(0, "Virus scanner failure: $esets_cli (error code: $errval)");
}
}
```

Note that the above script accepts the email only in case it is accepted in **esets_cli**'s Handle Object Policy. In any other case, the mail is blocked. If it a virus was found, it's name is extracted from the output.

Next, if you are using the Linux *RSR* package, you have to update your `PATH` environment variable with this command:

```
export PATH="$PATH:/opt/eset/esets/bin"
```

For successful installation you may need to install additional software like `arc`, `unarj`, `unrar`,

zoo. You also have to make a symlink in /usr/bin from uncompress to gzip and create the user amavis in group amavis with home dir /var/amavis. Now continue with the usual installation process (./configure, make, make install) and follow the rules README.mta according your mail server.

5.5.1.2. amavisd

Configuration of Amavisd is performed during the process of Amavisd installation. Unpack the source amavisd-0.x.tgz and follow the rules for amavis described in previous section of this guide. After 'make install' you may need to move '/usr/etc/amavisd.conf' to '/etc' and do a 'make install' again.

5.5.1.3. amavisd-new

In order to install the product with Amavisd-new, unpack and install the source amavisd-new-2.x.y.tgz in your installation directory. Now to configure the product with newly installed Amavisd-new, delete the clause for 'ESET Software ESETS' and replace the clause for 'ESET Software ESETS - Client/Server Version' in file 'amavisd.conf' with the following one:

```
### http://www.eset.com/
['ESET Software ESETS Command Line Interface',
 '@BINDIR@/esets_cli', '--subdir {}',
 [0], [1], qr/virus="([\^]+)"/ ],
```

You may need to install additional Perl modules Archive-Tar, Archive-Zip, BerkeleyDB, Compress-Zlib, Convert-TNEF, Convert-UUlib, IO-stringy, MailTools, MIME-Base64, MIME-tools, Net-Server and Unix-Syslog from www.cpan.org/modules. The procedure is by each as follows: perl Makefile.PL; make; make install.

After configuration, please follow the recommendation for configuring Amavisd-new in README.mta located in Amavisd-new directory according your mail server.

5.5.2. Scanning e-mail messages in KerioMailServer

- Install ESET Mail Security package, import the license file and make sure it gets regularly updated by `esets_daemon` or e.g. `cron`.
- Connect to Kerio MailServer using Kerio Administrator Console.
- In the Configuration > Content Filter > Antivirus menu check „Use external antivirus“ and select „NOD32 for Linux“.
- Click on the „Options“ button and update the paths for non-rsr version to:

```
LicenseDirectory: /etc/esets/license
NodDll: /usr/lib/libesets.so
NodModulesPath: /var/lib/esets
TmpDirectory: /tmp
```

for rsr-version:

```
LicenseDirectory: /etc/opt/eset/esets/license
NodDll: /opt/eset/esets/lib/libesets.so
NodModulesPath: /var/opt/eset/esets/lib
TmpDirectory: /tmp
```

- Confirm changes by pressing „Apply“.

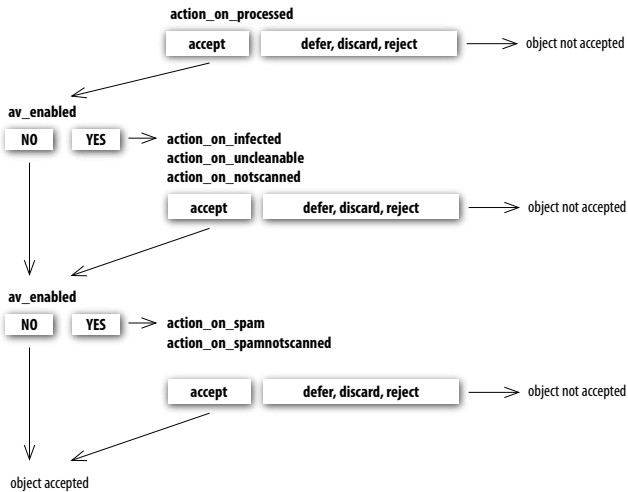
Chapter 6:

Important ESET Mail Security mechanisms

6.1. Handle Object Policy

The Handle Object Policy (see figure 5-1) is a mechanism that provides handling of the scanned objects depending on their scanning status. The mechanism is based on so-called action configuration options ('action_on_processed', 'action_on_infected', 'action_on_uncleanable', 'action_on_notscanned', 'action_on_spam', 'action_on_spamnotscanned') combined with Anti-Virus and Anti-Spam enabling configuration options ('av_enabled', 'as_enabled'). For detailed information on the options, please refer to the esets.cfg(5) manual page.

Figure 6-1. Scheme of Handle Object Policy mechanism.



Every object processed is at first handled with respect to the setting of the configuration option 'action_on_processed'. Once the option is set to 'accept', the object is handled according to the setting of configuration option 'av_enabled'. Once 'av_enabled' is enabled the object is scanned for virus infiltrations and set of action configuration options 'action_on_infected', 'action_on_uncleanable' and 'action_on_notscanned' is taken into account to evaluate further handling of the object. If action 'accept' has been taken as a result of the three above action options or 'av_enabled' is disabled the object processed shall be scanned for spam.

Note that object is scanned for spam only in case the configuration option 'as_enabled' is enabled. In this case the action configuration options 'action_on_spam' and 'action_on_spamnotscanned' is taken into account. If action 'accept' has been taken as a result of the two above action options or 'as_enabled' is disabled the object is accepted for further delivery, otherwise the object is blocked and is handled according to the particular action taken.

6.2. User Specific Configuration

User Specific Configuration mechanism is implemented in the product in order to provide administrator with enhanced configuration functionality. It allows to define *ESETS* anti-virus scanner parameters selectively for client/server identification.

Please note that the detailed description of this functionality can be found in *esets.cfg(5)* manual page and manual pages referenced there. Thus in this section we will only provide short example of user specific configuration definition.

Let's say we use **esets_smtp** module as a content filter for MTA Postfix. The module is subjected to configuration section [smtp] in *ESETS configuration file*. The section is as follows:

```
[smtp]
agent_enabled = yes
listen_addr = "localhost"
listen_port = 2526
server_addr = "localhost"
server_port = 2525
action_on_processed = accept
```

In order to provide individual parameters setting one has to define 'user_config' parameter with the path to the special configuration file where the individual setting will be stored. In the next example we create reference to the special configuration file 'esets_smtp_spec.cfg' located within the *ESETS configuration directory*.

```
[smtp]
agent_enabled = yes
listen_addr = "localhost"
listen_port = 2526
server_addr = "localhost"
server_port = 2525
action_on_processed = accept
user_config = "esets_smtp_spec.cfg"
```

Once special configuration file referenced from within [smtp] section we have to create this file in the *ESETS configuration directory* and provide it with an appropriate individual settings. The next example shows individual setting of parameter 'action_on_processed' for recipient `rcptuser@rcptdomain.com`.

```
[rcptuser@rcptdomain.com]
action_on_processed = reject
```

Note that the section header name of the special section contains identification of the recipient for which we have created the individual setting. The section body then contains individual parameters specified for this identification. Thus with this special configuration all e-mails will be processed, i.e. scanned for infiltrations, with exception of the e-mails sent to `rcptuser@rcptdomain.com` that will be rejected without scanning.

6.3. Black-list and white-list

In the next example we demonstrate the black-list and also white-list creation for the **esets_smtp** configured as content filter for MTA Postfix. Note that we use configuration described in the previous section for this purpose.

Thus in order to create black-list used by **esets_smtp** we have to create the following group section within the special configuration file 'esets_smtp_spec.cfg' introduced in the previous section.

```
[black-list]
action_on_processed = reject
```

The next step is to add some SMTP server into the 'black-list' group. For this purpose we have to create special section

```
[|sndrname1@sndrdomain1.com]
parent_id = "black-list"
```

where 'sndrname1@sndrdomain1.com' is an e-mail address of the sender added into the 'black-list'. Note that with this setting all e-mail sent from this address will be rejected.

If we want to create the 'white-list' used by **esets_smtp** we have to create the following group section within the special configuration file 'esets_smtp_spec.cfg' introduced in the previous section.

```
[white-list]
action_on_processed = accept
av_enabled = no
as_enabled = no
```

Adding of sender's e-mail address into the list is self-explanatory.

Please, note the character '|' placed in front of the header name of the special section in case of sender address and not placed there in case of recipient address. To get description of the special header name syntax, please refer to the appropriate *ESETS* agent module manual page. For an **esets_smtp** please refer to esets_smtp(1) manual page.

6.4. Anti-Spam Control

The goal of anti-spam system is to filter all spam e-mail messages, i.e. the e-mail messages that the recipient users do not want, from data flow of the e-mail messages delivery process.

To get rid of spam, this product implements the anti-spam control mechanism. The anti-spam functionality can be enabled using parameter 'as_enabled' (to get description of the parameter see esets.cfg(5) manual page). Note that anti-spam scanning can be used only for e-mail objects, thus this functionality is relevant only for **esets_imap**, **esets_mda**, **esets_pipe**, **esets_pop3**, **esets_smtp** and **esets_smfi** modules.

Once anti-spam is enabled in any of the configuration sections the anti-spam scanning engine is initialized during the main scanning daemon start-up. During this process an appropriate anti-spam supporting modules are loaded from within the anti-spam cache directory.

It is also possible to configure anti-spam functionality using configuration file:

```
@ETC\DIR@/anti-spam/spamcatcher.conf
```

Note the number of files within this directory, each corresponding to different recommended settings of anti-spam engine. Note that the default configuration file corresponds to the configuration file 'spamcatcher.conf.faster'. In order to use any of the files just replace the default anti-spam configuration file 'spamcatcher.conf' with the chosen one and reload *ESETS daemon*.

6.5. Samples Submission System

Samples submission system is an intelligent ThreatSense.NET technology that provides catching of the infected objects found by advanced heuristics method and delivering these objects to the samples submission system server. All virus samples caught by the sample submission system will be processed by the team of ESET virus laboratory department and consequently added into the ESET virus database, if necessary.

NOTE: ACCORDING TO OUR LICENSE AGREEMENT: BY ENABLING SAMPLE SUBMISSION SYSTEM YOU ARE AGREEING TO ALLOW THE COMPUTER AND/OR PLATFORM ON WHICH THE **ESETS_DAEMON** IS INSTALLED TO COLLECT DATA WHICH MAY INCLUDE PERSONAL INFORMATION ABOUT YOU AND/OR THE USER OF THE COMPUTER AND SAMPLES OF NEWLY DETECTED VIRUSES OR OTHER THREATS AND SEND THEM TO OUR VIRUS LAB. THIS FEATURE IS TURNED OFF BY DEFAULT. WE WILL ONLY USE THIS INFORMATION AND DATA TO STUDY THE THREAT AND WILL TAKE REASONABLE STEPS TO PRESERVE THE CONFIDENTIALITY OF SUCH INFORMATION.

In order to turn on Samples Submission System, the samples submission system cache has to be initialized. This can be achieved by enabling configuration option 'samples_enabled' in [global] section of *ESETS configuration file*. In order to enable process of samples delivery to ESET virus laboratory servers it is yet necessary to enable parameter 'samples_send_enabled' in the same section.

User may decide to provide the ESET virus laboratory team with the additional optional information using configuration options 'samples_provider_mail' and/or 'samples_provider_country'. This information will help us to get overview on the infiltration spreading throughout the Internet.

In order to get detailed information on the Samples Submission System, refer to *esets_daemon(8)* manual page.

6.6. World WideWeb Interface

WWW Interface allows user-friendly ESETS configuration, administration and license management.

This module is a separate agent and must be explicitly enabled. For quickstart, set all of these options in *ESETS configuration file* and restart *ESETS daemon*:

```
[wwwi]
agent_enabled = yes
listen_addr = address
listen_port = port
username = name
password = pass
```

(enter all four values as your own ones) and direct your browser to 'https://address:port' (note the *https*) and login with 'name/pass'. There are basic usage instructions on the help page. For more technical details about **esets_wwwi** see the **esets_wwwi(1)** manual page.

Chapter 7:

ESET Mail Security system update

7.1. ESETS update utility

In order to keep the ESET Mail Security effective, it is necessary to keep its virus signatures database up to date. The `esets_update` utility has been developed for this purpose (see `esets_update(8)` manual page for details). In order to launch update one has to define configuration options 'username' and 'password' in [update] section of *ESETS configuration file*. Note that in case you access the Internet via HTTP proxy additional configuration options 'proxy_addr', 'proxy_port' and optionally 'proxy_username' and 'proxy_password' have to be specified there as well. To trigger an update, enter command:

```
@SBINDIR@/esets_update
```

To provide the highest security for the user, the ESET team collects the virus definitions continuously from all over the world. The new patterns can appear within the database in very short intervals. It is therefore recommended, to trigger an update on a regular basis. Note that *ESETS daemon* is able to provide the periodic update of the system once 'av_update_period' configuration option specified in [update] section of *ESETS configuration file* and the daemon is up and running.

7.2. ESETS update process description

The update process is composed of two stages. First, the mirror of all relevant so-called pre-compiled modules have to be created from the origin ESET server. The pre-compiled modules are downloaded by default into directory

```
@BASEDIR@/mirror
```

Note that the mirror directory path can be redefined using configuration option 'mirror_dir' in section [update] of *ESETS configuration file*.

The *ESETS* modules are divided into two categories; engine category and component category. The modules of component category are currently only for use on the MS Windows OS. Currently the following types of engine category modules are supported: base scanning modules (prefix engine) containing virus signatures database, archives support modules (prefix archs) supporting various file system archive formats, advanced heuristics modules (prefix advheur) containing implementation of so-called advanced heuristics method of virus and worm detection, packed worm scanner modules (prefix pwscan) used on MS Windows OS, utilities modules (prefix utilmod) used on MS Windows OS and ThreatSense.NET technology support modules (prefix charon). These modules are always necessary and therefore are all downloaded by default at each download process. On the other hand the component category modules are platform dependent and language localization dependent and thus the download of component category modules is optional.

After download of the pre-compiled modules the 'update.ver' file is created in the mirror directory as well. This file contains the information about the modules currently stored in the newly created mirror. The newly created mirror thus serves as fully functional modules download server and can be used to create subordinate mirrors, however, some more conditions have to be fulfilled yet. First, there must be a http server installed on the computer where the modules are going to be downloaded from. Second, the modules to be downloaded by other computers have to be placed at the directory path

```
/http-serv-base-path/nod_upd
```

where 'http-serv-base-path' is a base http server directory path, as this is the first place where update utility looks the modules for.

Second part of the update process is the compilation of modules loadable by the ESET Mail Security scanner from those stored in the local mirror. Typically the following *ESETS* loading modules are created: base module (nod32.000), archives support module (nod32.002), advanced heuristics module (nod32.003), packed worm scanner module (nod32.004), windows utilities module (nod32.005) and ThreatSense.NET support module (nod32.006) in the directory:

```
@BASEDIR@
```

Note that it is exactly the directory where *ESETS daemon* loads modules from and thus can be redefined by using configuration option 'base_dir' in section [global] (resp. [update]) of *ESETS configuration file*.





Chapter 8:

Tips and Tricks



8.1. ESETS and TLS support in MTA

Transport Layer Security (TLS) is a protocol guarantying data privacy in client/server communication over the Internet. The principle of TLS is based on the SSL encryption of data traveling between SMTP client and server. This has consequences for scanning the communication. Indeed, once TLS support in MTA is enabled, the 'wrapping' methods are impossible as the whole intercepted SMTP communication is encrypted at this stage. On the other hand, there is possibility to use data encryption in communication between local MTA and Internet and still use the 'content filtering' methods. In MTA Sendmail there is no problem with SMTP TLS support at all as the content filtering is done internally using Milter. On the other hand the Postfix uses SMTP protocol for data communication between content filter and MTA. Therefore once the TLS is enabled in Postfix, the content filtering method fails as communication is encrypted.

Fortunately, this can be solved on the Postfix TLS configuration level by deactivation of the TLS support for communication between client and server within localhost. Add the following line into `/etc/postfix/main.cf`:

```
smtp_tls_per_site = hash:/etc/postfix/smtp_tls_per_site
```

In addition you have to create `/etc/postfix/smtp_tls_per_site` file with the following content:

```
localhost      NONE
```

and provide its appropriate hash table by entering the following command from `/etc/postfix/` directory:

```
postmap hash:smtp_tls_per_site
```

By using the above statement the `/etc/postfix/smtp_tls_per_site.db` file is created that is used by Postfix to enable TLS on per site basis. As far as we have disabled TLS for localhost the content filtering can be used and at the same time the SMTP communication between local MTA and Internet is encrypted.



Chapter 9:

Let us know



Dear user, this guide should have given you a good knowledge about the ESET File Security installation, configuration and maintenance. However, writing a documentation is a process that is never finished. There will always be some parts that can be explained better or are not even explained at all. Therefore, in case of bugs or inconsistencies found within this documentation, please report a problem to our support center

<http://www.nod32.com.hk/support>

We are looking forward to help you solve any problem concerning the product.



Appendix A. *ESETS* setup process description

A.1. Setting *ESETS* for MTA Postfix

A.1.1. Inbound e-mail messages scanning

Warning: This installation is not compatible with SELinux. Either disable SELinux or follow the next section.

The goal of this installation is to insert **esets_mda** before Postfix original MDA. Used MDA (with arguments) is set in the Postfix parameter 'mailbox_command'.

Note: If the value is empty, Postfix is delivering mail by himself. You have to install and configure a real MDA (e.g. procmail) and use that first for 'mailbox_command' including arguments (e.g. /usr/bin/procmail -d "\$USER"). Reload Postfix and make sure it is delivering mail according your needs. Now you can continue with *ESETS* installation.

Take the full path to the current Postfix MDA and set the parameter 'mda_path' in [mda] section of *ESETS configuration file* to this value, in our sample case:

```
mda_path = "/usr/bin/procmail"
```

and restart *ESETS daemon*. Then replace the path to current Postfix MDA with **esets_mda** path and add --recipient="\$RECIPIENT" --sender="\$SENDER" to the arguments, in our sample case:

```
mailbox_command = @BINDIR@/esets_mda -d "$USER"  
-- --recipient="$RECIPIENT" --sender="$SENDER"
```

For reread of newly created configuration, reload Postfix.

A.1.2. Bi-directional e-mail messages scanning

The goal of this installation is to divert all mails from Postfix to **esets_smtp** and get them back. In the [smtp] section of *ESETS configuration file* set these parameters:

```
agent_enabled = yes  
listen_addr = "localhost"  
listen_port = 2526  
server_addr = "localhost"  
server_port = 2525
```

and restart *ESETS daemon*. It will start **esets_smtp** and make it scan all SMTP communication accepted on 'listen_addr:listen_port' and forward it to 'server_addr:server_port'. To divert all mail to **esets_smtp** set in Postfix:

```
content_filter = smtp:[127.0.0.1]:2526
```

Note: In case the 'content_filter' parameter already has a value, don't follow these instructions. Instead you have to insert **esets_smtp** (or other *ESETS* mail scanning module) before or after your current 'content_filter'.

The last thing is to make Postfix accept mail on port 2525 and continue processing it. Add this entry to Postfix master.cf file:

```
localhost:2525 inet n - n - - smtpd
  -o content_filter=
  -o myhostname=esets.yourdomain.com
  -o local_recipient_maps=
  -o relay_recipient_maps=
  -o receive_override_options=no_unknown_recipient_checks,no_header_body_checks
  -o smtpd_helo_restrictions=
  -o smtpd_client_restrictions=
  -o smtpd_sender_restrictions=
  -o smtpd_recipient_restrictions=permit_mynetworks,reject
  -o mynetworks=127.0.0.0/8
```

just replace yourdomain.com with your hostname after first dot. Make sure all but the first line is indented. For reread of newly created configuration, reload Postfix.

Note: In case you have SELinux enabled which prevents Postfix to listen on ²⁵²⁵ (eg. Fedora Core >= 5), run this command: `semanage -a t smtp_port_t p tcp 2525`

A.2. Setting *ESETS* for MTA Sendmail

A.2.1. Inbound e-mail messages scanning

Warning: This installation is not compatible with SELinux. Either disable SELinux or follow the next section.

The goal of this installation is to insert **esets_mda** before Sendmail's original MDA.

Note: On FreeBSD, Sendmail may be communicating with MDA using LMTP. However **esets_mda** does not understand LMTP. So if you have `FEATURE{local_lmtp}` in `'hostname'.mc` comment it out now and recreate `sendmail.cf`.

The currently used MDA can be found in the file `sendmail.cf` in section `Mlocal: parameters 'P' (executable) and 'A' (its name and arguments)`.

First set `'mda_path'` in `[mda]` section of *ESETS configuration file* to the currently used MDA executable (Sendmail's `'P'` parameter) and restart *ESETS daemon*.

Then add to file `sendmail.mc` (or `'hostname'.mc` on FreeBSD) before all MAILER definitions these lines:

```
define(`LOCAL_MAILER_PATH', `@BINDIR@/esets_mda')dnl
define(`LOCAL_MAILER_ARGS',
  `esets_mda original_arguments -- --sender $f --recipient $u@$j')dnl
```

where `original_arguments` is Sendmail's `'A'` parameter without the name (first word).

In the last, recreate `sendmail.cf` and restart Sendmail.

A.2.2. Bi-directional e-mail messages scanning

The goal of this installation is to scan all mails in Sendmail with **esets_smfi** filter. In the [smfi] section of *ESETS configuration file* set these parameters:

```
agent_enabled = yes
smfi_sock_path = "/var/run/esets_smfi.sock"
```

and restart *ESETS daemon*. Then add to file *sendmail.mc* (or *hostname.mc* on FreeBSD) before all MAILER definitions this line:

```
INPUT_MAIL_FILTER(`esets_smfi',
`S=local:/var/run/esets_smfi.sock, F=T, T=S:2m;R:2m;E:5m') dnl
```

With these settings, Sendmail will communicate with **esets_smfi** via unix socket */var/run/esets_smfi.sock*. Flag *F=T* will result in a temporary fail connection, if the filter is unavailable. Timeouts *S:2m* defines 2 minutes timeout for sending information from MTA to filter, *R:2m* defines 2 minutes timeout for reading reply from the filter and *E:5m* means overall 5 minutes timeout between sending end-of-message to filter and waiting for the final acknowledgment.

Note that in case the timeouts for the **esets_smfi** filter are set too small, Sendmail can temporarily defer the message to the queue and attempt to pass it through later. This may lead to continuous deferral of the same messages. In order to avoid the problem, the timeouts have to be set properly. One can also experiment with the Sendmail's *'confMAX_MESSAGE_SIZE'* parameter, which is the maximum accepted message size in bytes. Taking into account this value and the maximum time for processing of this amount of data by MTA (this can be measured), one can evaluate the appropriate timeouts for **esets_smfi** filter.

In the last, recreate *sendmail.cf* and restart Sendmail.

A.3. Setting *ESETS* for MTA Qmail

A.3.1. Inbound e-mail messages scanning

The goal of this installation is to insert **esets_mda** before Qmail's local delivery agent. Let's assume, Qmail is installed in the */var/qmail* directory. In the [mda] section of *ESETS configuration file* set this parameter:

```
mda_path = "/var/qmail/bin/qmail-esets_mda"
```

and restart *ESETS daemon*. Create the file */var/qmail/bin/qmail-esets_mda* with this content and run *chmod a+x* on it:

```
#!/bin/sh
exec qmail-local -- "$USER" "$HOME" "$LOCAL" "" "$EXT" \
"$HOST" "$SENDER" "$1"
```

which will make **esets_mda** call Qmail's local delivery agent. Now create the file */var/qmail/bin/qmail-start.esets* with this content and also run *chmod a+x* on it:

```
#!/bin/sh
A="$1"; shift
```

```
exec qmail-start.orig "|@BINDIR@/esets_mda `\$A'"" \
-- --sender="\$SENDER" --recipient="\$RECIPIENT"" `\$@"
```

which will start Qmail using **esets_mda** for local deliveries. However, the original delivery specification is passed to qmail-local through **esets_mda**. Note that in this configuration **esets_mda** will use Qmail's recognized exit codes (see qmail-command(8)). Lastly, replace qmail-start using commands:

```
mv /var/qmail/bin/qmail-start /var/qmail/bin/qmail-start.orig
ln -s qmail-start.esets /var/qmail/bin/qmail-start
```

and restart Qmail.

A.3.2. Bi-directional e-mail messages scanning

The goal of this installation is to insert **esets_mda** before qmail-queue, which queues all mails before delivery. Let's assume Qmail is installed in the /var/qmail directory. In the [mda] section of *ESETS configuration file* set this parameter:

```
mda_path = "/var/qmail/bin/qmail-queue.esets"
```

and restart *ESETS daemon*. Lastly, replace qmail-queue using commands:

```
mv /var/qmail/bin/qmail-queue /var/qmail/bin/qmail-queue.esets
ln -s @BINDIR@/esets_mda /var/qmail/bin/qmail-queue
```

No Qmail restarting is necessary. All messages enqueued from now will be scanned by *ESETS*. Note that in this configuration **esets_mda** will use qmail-queue's exit codes (see qmail-queue(8)).

A.4. Setting *ESETS* for MTA Exim version 3

A.4.1. Inbound e-mail messages scanning

The goal of this installation is to create an Exim transport from **esets_mda** for local users. In the [mda] section of *ESETS configuration file* set this parameter:

```
mda_path = "/usr/sbin/exim"
```

where /usr/sbin/exim is the full path to Exim binary. Then restart *ESETS daemon*. Next, add this transport (at whatever place) to the list of Exim transports:

```
esets_transport:
  driver = pipe
  command = @BINDIR@/esets_mda -oi -oMr esets-scanned $local_part@$domain \
  -- --sender=$sender_address --recipient=$local_part@$domain
  user = mail
```

where mail is one of Exim's 'trusted_users'. Then add this director as first to the list of Exim directors:

```
esets_director:
```

```
driver = smartuser
condition = "${if eq {$received_protocol}{esets-scanned} {0}{1}}"
transport = esets_transport
verify = false
```

which will send all not-yet-scanned mails for local users to **esets_mda**, which will inject them back to Exim for further processing. For reread of newly created configuration, restart Exim.

A.4.2. Bi-directional e-mail messages scanning

The goal of this installation is to create an Exim transport from **esets_mda** for all mails. Perform all steps from the previous section, but also add this router as first to the list of Exim routers:

```
esets_router:
driver = domainlist
route_list = "* localhost byname"
condition = "${if eq {$received_protocol}{esets-scanned} {0}{1}}"
transport = esets_transport
verify = false
```

A.5. Setting *ESETS* for MTA Exim version 4

A.5.1. Inbound e-mail messages scanning

The goal of this installation is to create an Exim transport from **esets_mda** for local users. In the [mda] section of *ESETS configuration file* set this parameter:

```
mda_path = "/usr/sbin/exim"
```

where `/usr/sbin/exim` is the full path to Exim binary. Then restart *ESETS daemon*. Add this router as first to the list of Exim routers:

```
esets_router:
driver = accept
domains = +local_domains
condition = "${if eq {$received_protocol}{esets-scanned} {0}{1}}"
transport = esets_transport
verify = false
```

and this transport (at whatever place) to the list of Exim transports:

```
esets_transport:
driver = pipe
command = @BINDIR/esets_mda -oi -oMr esets-scanned $local_part@$domain \
-- --sender=$sender_address --recipient=$local_part@$domain
```

which will send all not-yet-scanned mails for local users to **esets_mda**, which will inject them back to Exim for further processing. For reread of newly created configuration, restart Exim.

A.5.2. Bi-directional e-mail messages scanning

The goal of this installation is to create an Exim transport from **esets_mda** for all mails. Perform all steps from the previous section, but omit this line in **esets_router**:

```
domains = +local_domains
```

A.6. Setting *ESETS* for outbound e-mail messages scanning

The outbound e-mail messages scanning is performed using **esets_smtp** daemon. In the [smtp] section of *ESETS configuration file* set these parameters:

```
agent_enabled = yes
listen_addr = "192.168.1.0"
listen_port = 2525
```

where 'listen_addr' is the address of local network interface named *if0*. Then restart *ESETS daemon*. The next step is to redirect all SMTP requests to **esets_smtp**. In case of IP-filtering provided by ipchains administration tool an appropriate rule is:

```
ipchains -A INPUT -p tcp -i if0 --dport 25 -j REDIRECT 2525
```

If IP-filtering mechanism is provided by iptables administration tool, the rule is:

```
iptables -t nat -A PREROUTING -p tcp -i if0 \
--dport 25 -j REDIRECT --to-ports 2525
```

resp. if ipfw tool used (in case BSD OS) the rule is as follows:

```
ipfw add fwd 192.168.1.10,2525 tcp from any to any 25 via if0 in
```

Warning: Your MTA may accept all connections without extensive checking from **esets_smtp** because they are local. By using your own firewall rules, make sure you do not create an open relay, i.e. allow someone from the outside to connect to **esets_smtp** and thus use him as relay SMTP server.

A.7. Setting *ESETS* for scanning of POP3 communication

The POP3 communication scanning is performed using **esets_pop3** daemon. In the [pop3] section of *ESETS configuration file* set these parameters:

```
agent_enabled = yes
listen_addr = "192.168.1.10"
listen_port = 8110
```

where 'listen_addr' is the address of local network interface named *if0*. Then restart *ESETS daemon*. The next step is to redirect all POP3 requests to **esets_pop3**. In case of IP-filtering provided by ipchains administration tool an appropriate rule is:

```
ipchains -A INPUT -p tcp -i if0 --dport 110 -j REDIRECT 8110
```

If IP-filtering mechanism is provided by iptables administration tool, the rule is:

```
iptables -t nat -A PREROUTING -p tcp -i if0 \  
--dport 110 -j REDIRECT --to-ports 8110
```

resp. if ipfw tool used (in case BSD OS) the rule is as follows:

```
ipfw add fwd 192.168.1.10,8110 tcp from any to any 110 via if0 in
```

A.8. Setting *ESETS* for scanning of IMAP communication

The IMAP communication scanning is performed using *esets_imap* daemon. In the [imap] section of *ESETS configuration file* set these parameters:

```
agent_enabled = yes  
listen_addr = "192.168.1.10"  
listen_port = 8143
```

where 'listen_addr' is the address of local network interface named *if0*. Then restart *ESETS daemon*. The next step is to redirect all IMAP requests to *esets_imap*. In case of IP-filtering provided by ipchains administration tool an appropriate rule is:

```
ipchains -A INPUT -p tcp -i if0 --dport 143 -j REDIRECT 8143
```

If IP-filtering mechanism is provided by iptables administration tool, the rule is:

```
iptables -t nat -A PREROUTING -p tcp -i if0 \  
--dport 143 -j REDIRECT --to-ports 8143
```

resp. if ipfw tool used (in case BSD OS) the rule is as follows:

```
ipfw add fwd 192.168.1.10,8143 tcp from any to any 143 via if0 in
```



Appendix A. PHP License



The PHP License, version 3.01 Copyright (c) 1999 - 2006 The PHP Group. All rights reserved. Redistribution and use in source and binary forms, with or without modification, is permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "PHP" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact group@php.net.
4. Products derived from this software may not be called "PHP", nor may "PHP" appear in their name, without prior written permission from group@php.net. You may indicate that your software works in conjunction with PHP by saying "Foo for PHP" instead of calling it "PHP Foo" or "phpfoo".
5. The PHP Group may publish revised and/or new versions of the license from time to time. Each version will be given a distinguishing version number. Once covered code has been published under a particular version of the license, you may always continue to use it under the terms of that version. You may also choose to use such covered code under the terms of any subsequent version of the license published by the PHP Group. No one other than the PHP Group has the right to modify the terms applicable to covered code created under this License.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes PHP software, freely available from <http://www.php.net/software/>".

THIS SOFTWARE IS PROVIDED BY THE PHP DEVELOPMENT TEAM ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE PHP DEVELOPMENT TEAM OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.