ESET File Security

Installation Manual and User Guide

Linux, BSD and Solaris

ESET File Security

Customer Care Worldwide: www.eset.eu/support
Customer Care North America: www.eset.com/support

REV. 2011-02-08

Contents

1. Introduction

Dear user, you have acquired ESET File Security - the premier security system running under the Linux, BSD and Solaris OS. As you will soon find out, ESET's state-of-the-art scanning engine has unsurpassed scanning speed and detection rates combined with a very small footprint that makes it the ideal choice for any Linux, BSD and Solaris OS server.

1.1 Main functionality

On-demand scanner

The On-demand scanner can be invoked by a privileged user (usually a system administrator) through either the command line interface or the web interface; or by the operating system's automatic scheduling tool (e.g., cron). Thus, the term *On-demand* refers to file system objects being scanned on user or system demand.

On-access scanner

The On-access scanner is invoked whenever a user and/or operating system attempts to access file system objects. This also clarifies the use of the term *On-access*; because a scan is triggered by any attempt to access file system objects.

1.2 Key features of the system

Advanced engine algorithms

The ESET antivirus scanning engine algorithms provide the highest detection rate and the fastest scanning times.

Multi-processing

ESET File Security is developed to run on single- as well as multi-processor units.

Advanced Heuristics

ESET File Security includes unique advanced heuristics for Win32 worms, backdoor infections and other forms of malware.

Built-In features

Built-in archivers unpack archived objects without the need for any external programs.

Speed and efficiency

To increase the speed and efficiency of the system, its architecture is based on the running daemon (resident program) where all scanning requests are sent.

Enhanced security

All executive daemons (except esets_dac) run under non-privileged user account to enhance security.

Selective configuration

The system supports selective configuration based on the user or client/server.

Multiple logging levels

Multiple logging levels can be configured to get information about system activity and infiltrations.

Web interface

Configuration, administration and license management are offered through an intuitive and user-friendly Web interface.

Remote administration

The system supports ESET Remote Administration for management in large computer networks.

No external libraries

The ESET File Security installation does not require external libraries or programs except for LIBC.

User-specified notification

The system can be configured to notify specific users in the event of a detected infiltration or other important events.

Low system requirements

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

Performance and scalability

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

2. Terminology and abbreviations

In this section we will review the terms and abbreviations used in this document. Note that a boldface font is reserved for product component names and also for newly defined terms and abbreviations. Terms and abbreviations defined in this chapter are expanded upon later in this document.

ESETS

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

RSR

Abbreviation for 'RedHat/Novell(SuSE) Ready'. Note that we also support RedHat Ready and Novell(SuSE) Ready variations of the product. The RSR package differs from the 'standard' Linux version in that it meets the FHS (File-system Hierarchy Standard defined as a part of Linux Standard Base) criteria required by the RedHat Ready and Novell(SuSE) Ready certificate. This means that the RSR package is installed as an add-on application - the primary installation directory is '/opt/eset/esets'.

ESETS daemon

The main ESETS system control and scanning daemon: *esets_daemon*.

ESETS base directory

The directory where ESETS loadable modules containing the virus signature database are stored. The abbreviation *@BASEDIR@* will be used for future references to this directory. The *@BASEDIR@* value for the following Operating Systems is listed below:

```
Linux: /var/lib/esets
Linux RSR: /var/opt/eset/esets/lib
FreeBSD: /var/lib/esets
NetBSD: /var/lib/esets
Solaris: /var/opt/esets/lib
```

ESETS configuration directory

The directory where all files related to the ESET File Security configuration are stored. The abbreviation *@ETCDIR@* will be used for future references to this directory. The *@ETCDIR@* value for the following Operating Systems is listed below:

```
Linux: /etc/esets
Linux RSR: /etc/opt/eset/esets
FreeBSD: /usr/local/etc/esets
NetBSD: /usr/pkg/etc/esets
Solaris: /etc/opt/esets
```

ESETS configuration file

Main ESET File 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 File Security binary files are stored. The abbreviation *@BINDIR@* will be used for future references to this directory. The *@BINDIR@* value for the following Operating Systems is listed below:

```
Linux: /usr/bin
Linux RSR: /opt/eset/esets/bin
FreeBSD: /usr/local/bin
NetBSD: /usr/pkg/bin
Solaris: /opt/esets/bin
```

ESETS system binary files directory

The directory where the relevant ESET File Security system binary files are stored. The abbreviation *@SBINDIR@* will be used for future references to this directory. The *@SBINDIR@* value for the following Operating Systems is listed below:

```
Linux: /usr/sbin
Linux RSR: /opt/eset/esets/sbin
FreeBSD: /usr/local/sbin
NetBSD: /usr/pkg/sbin
Solaris: /opt/esets/sbin
```

ESETS object files directory

The directory where the relevant ESET File Security object files and libraries are stored. The abbreviation *@LIBDIR@* will be used for future references to this directory. The *@LIBDIR@* value for the following Operating Systems is listed below:

```
Linux: /usr/lib/esets
Linux RSR: /opt/eset/esets/lib
FreeBSD: /usr/local/lib/esets
NetBSD: /usr/pkg/lib/esets
Solaris: /opt/esets/lib
```

3. Installation

After purchasing ESET File Security, you will receive your authorization data (username, password and license key). This data is necessary for both identifying you as our customer and allowing you to download updates for ESET File Security. The username/password data is also required for downloading the initial installation package from our web site. ESET File Security is distributed as a binary file:

```
esets.i386.ext.bin
```

In the binary file shown above, *'ext'* is a Linux, BSD and Solaris 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.x, 'fbs6.tgz' for FreeBSD 6.x, 'nbs4.tgz' for NetBSD 4.xx and 'sol10.pkg.gz' for Solaris 10.

Note that the Linux RSR binary file format is:

```
esets-rsr.i386.rpm.bin
```

To install or upgrade the product, use the following command:

```
sh ./esets.i386.ext.bin
```

For the Linux RSR variation of the product, use the command:

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

to display the product's User License Acceptance Agreement. 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 upgrade is displayed onscreen.

Once the package is installed, you can verify that the main ESETS service is running by using the following command:

Linux OS:

```
ps -C esets_daemon
```

BSD OS:

```
ps -ax | grep esets_daemon
```

Solaris:

```
ps -A | grep esets_daemon
```

After pressing ENTER, you should see the following (or similar) message:

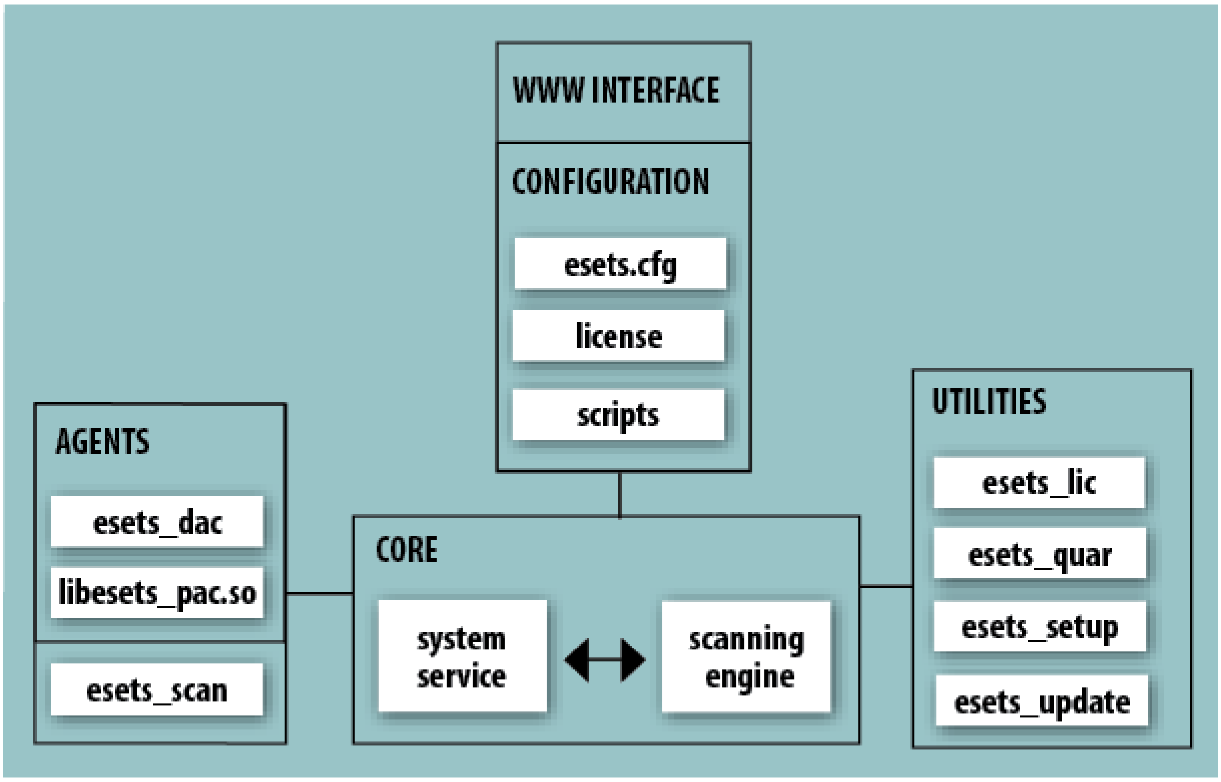
```
 PID TTY          TIME CMD
2226 ?        00:00:00 esets_daemon
2229 ?        00:00:00 esets_daemon
```

At least two ESETS daemon processes are running in the background. The first PID represents the process and threads manager of the system. The other represents the ESETS scanning process.

4. Architecture Overview

Once ESET File Security is successfully installed, you should become familiar with its architecture.

Figure 4-1. Structure of ESET File Security.



The structure of ESET File Security is shown in Figure 4-1. The system is comprised of the following parts:

CORE

The Core of ESET File Security is the ESETS daemon (esets_daemon). The daemon uses ESETS API library libesets.so and ESETS loading modules em00X_xx.dat to provide base system tasks such as scanning, maintenance of the agent daemon processes, maintenance of the samples submission system, logging, notification, etc. Please refer to the *esets_daemon(8)* man page for details.

AGENTS

The purpose of ESETS agent modules is to integrate ESETS with the Linux, BSD and Solaris Server environment.

UTILITIES

The utility modules provide simple and effective management of the system. They are responsible for relevant system tasks such as license management, quarantine management, system setup and update.

CONFIGURATION

Proper configuration is the most important aspect of a smooth-running security system - the remainder of this chapter is dedicated to explaining all related components. A thorough understanding of the *esets.cfg* file is also highly recommended, as this file contains information essential to the configuration of ESET File Security.

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

@ETCDIR@/esets.cfg

This is the most important configuration file, as it controls all major aspects of the product‘s functionality. The esets.cfg file is made up of several sections, each of which contains various parameters. The file contains one global and several “agent“ sections, with all section names enclosed in square brackets. Parameters in the global section are used to define configuration options for the ESETS daemon as well as default values for the ESETS scanning engine configuration. Parameters in agent sections are used to define configuration options of modules used to intercept various data flow types in the computer and/or its neighborhood, and prepare it for scanning. Note that in addition to the various parameters used for system configuration, there are also rules governing the organization of the file. For detailed information on the most effective way to organize this file, please refer to the *esets.cfg(5)* and *esets_daemon(8)* man pages, as well as relevant agents' man pages.

@ETCDIR@/certs

This directory is used to store the certificates used by the ESETS web interface for authentication. Please see the *esets_wwwi(8)* man page 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 check only this directory for a valid license key, unless the *'license_dir'* parameter in the ESETS configuration file is redefined.

@ETCDIR@/scripts/license_warning_script

If enabled by the ESETS configuration file parameter *'license_warn_enabled'*, this script will be executed 30 days (once per day) before product license expiration, sending an email notification about the expiration status to the system administrator.

@ETCDIR@/scripts/daemon_notification_script

If enabled by the ESETS configuration file parameter *'exec_script'*, this script is executed in the event of a detected infiltration by the antivirus system. It is used to send email notification about the event to the system administrator.

5. Integration with File System services

This chapter describes the On-demand and On-access scanner configuration which will provide the most effective protection from virus and worm file system infections. ESET File Security's scanning power is derived from the On-demand scanner command *'esets_scan'* and the On-access scanner command *'esets_dac'*. The Linux version of ESET File Security offers an additional On-access scanner technique which uses the preloaded library module *libesets_pac.so*. All of these commands are described in the following sections.

Warning! Novell Storage Services (NSS) break common unix security principles the scanner relies on when limiting privileges. This results in no threat detection on NSS mounted volumes. If you have such mounted volume, set the *'esets_user'* parameter to *'root'* in ESETS configuration file and restart ESETS daemon.

5.1 On-demand scanner

The On-demand scanner can be invoked by a privileged user (usually a system administrator) through the command line interface or web interface, or by the operating system's automatic scheduling tool (e.g., cron). Thus, the term *On-demand* refers to file system objects which are scanned on user or system demand.

The On-demand scanner does not require special configuration in order to run. After the ESETS package has been properly installed and a valid license has been moved to the license keys directory (@ETCDIR@/license), the On-demand scanner can be run immediately using the command line interface or the Scheduler tool. To run the On-demand scanner from the command line, use the following syntax:

```
@SBINDIR@/esets_scan [option(s)] FILES
```

where FILES is a list of directories and/or files to be scanned.

Multiple command line options are available using ESETS On-demand scanner. To see the full list of options, please see the *esets_scan(8)* man page.

5.2 On-access scanner powered by Dazuko

The On-access scanner is invoked by user(s) access and/or operating system access to file system objects. This also explains the term *On-access*; the scanner is triggered on any attempt to access a selected file system object.

The technique used by ESETS On-access scanner is powered by the Dazuko (da-tzu-ko) kernel module and is based on the interception of kernel calls. The Dazuko project is open source, which means that its source code is freely distributed. This allows users to compile the kernel module for their own custom kernels. Note that the Dazuko kernel module is not a part of any ESETS product and must be compiled and installed into the kernel prior to using the On-access command *esets_dac*. On the other hand the Dazuko technique makes On-access scanning independent from the file system type used. It is also suitable for scanning of file system objects via Network File System (NFS), Nettalk and Samba.

Important: Before we provide detailed information related to On-access scanner configuration and use, it should be noted that the scanner has been primarily developed and tested to protect externally mounted file systems. In case of multiple file systems that are not externally mounted, you will need to exclude them from file access control in order to prevent system hang ups. An example of a typical directory to exclude is the *'/dev'* directory and any directories used by ESETS.

5.2.1 Operation principle

The On-access scanner *esets_dac* (ESETS Dazuko-powered file Access Controller) is a resident program which provides continuous monitoring and control over the file system. Every file system object is scanned based on customizable file access event types. The following event types are supported by the current version:

Open events

To activate this file access type set the value of the *'event_mask'* parameter to open in the ***[dac]*** section of the esets.cfg file. This will enable the ON_OPEN bit of the Dazuko access mask.

Close events

To activate this file access type set the value of the *'event_mask'* parameter to close in the ***[dac]*** section of the esets.cfg file. This will enable the ON_OPEN bit of the Dazuko access mask. This will enable the ON_CLOSE and ON_CLOSE_MODIFIED bits of the Dazuko access mask.

NOTE: Some OS kernel versions do not support the interception of ON_CLOSE events. In these cases, close events will not be monitored by *esets_dac*.

Exec events

To activate this file access type set the value of the *'event_mask'* parameter to exec in the ***[dac]*** section of the esets.cfg file. This will enable the ON_EXEC bit of the Dazuko access mask.

The On-access scanner ensures that all opened, closed and executed files are first scanned by the esets_daemon for viruses. Depending on the scan results, access to specific files is denied or allowed.

5.2.2 Installation and configuration

The Dazuko kernel module must be compiled and installed within the running kernel before initializing *esets_dac*. For details on how to compile and install Dazuko, please see:

http://www.dazuko.org

Once Dazuko is installed, review and edit the ***[global]*** and ***[dac]*** sections of the ESETS configuration file (esets.cfg). Note that proper functioning of the On-access scanner is dependent upon configuration of the *'agent_enabled'* option within the ***[dac]*** section of this file. Additionally, you must define the file system objects (i.e. directories and files) that are to be monitored by the On-access scanner. This can be accomplished by defining the parameters of the *'ctl_incl'* and *'ctl_excl'* options, which are also located within the ***[dac]*** section. After making changes to the esets.cfg file, you can force the newly created configuration to be re-read by reloading the ESETS daemon.

5.2.3 Tips

To ensure that the Dazuko module loads prior to initialization of the *esets_dac* daemon, follow these steps:

Place a copy of the Dazuko module in either of the following directories reserved for kernel modules:

```
/lib/modules
```

or

```
/modules
```

Use the kernel utilities 'depmod' and 'modprobe' (For BSD OS, use 'kldconfig' and 'kldload') to handle dependencies and successful initialization of the newly added Dazuko module.

In the esets_daemon initialization script '/etc/init.d/esets_daemon', insert the following line before the daemon initialization statement:

```
/sbin/modprobe dazuko
```

For BSD OS's the line

```
/sbin/kldconfig dazuko
```

must be inserted into the '/usr/local/etc/rc.d/esets_daemon.sh' script.

Warning! It is extremely important that these steps are executed in the exact order given. If the kernel module is not located within the kernel modules directory it will not properly load, causing system hang-ups.

5.3 On-access scanner using preload LIBC library

In the previous sections we described the integration of the On-access scanner powered by Dazuko with Linux/BSD file system services. If, however, the use of Dazuko is not feasible, for example for system administrators who maintain critical systems where:

- the source code and/or configuration files related to the running kernel are not available,
- the kernel is more monolithic than modular,
- the Dazuko module simply does not support the given OS.

In any of these cases, the On-access scanning technique based on the preload LIBC library should be used. See the following topics in this section for detailed information. Please note that this section is relevant only for Linux OS users and contains information regarding the operation, installation and configuration of the On-access scanner using the preload library *'libesets_pac.so'*.

5.3.1 Operation principle

The On-access scanner *libesets_pac.so* (ESETS Preload library based file Access Controller) is a shared objects library which is activated at system start-up. This library is used for LIBC calls by file system servers such as FTP server, Samba server etc. Every file system object is scanned based on customizable file access event types. The following event types are supported by the current version:

Open events

This file access type is activated if the word *'open'* is present in the *'event_mask'* parameter in the esest.cfg file (***[pac]*** section).

Close events

This file access type is activated if the word *'close'* is present in the *'event_mask'* parameter in the esets.cfg file (***[pac]*** section). In this case, all file descriptor and FILE stream close functions of the LIBC are intercepted.

Exec events

This file access type is activated if the word *'exec'* is present in the *'event_mask'* parameter in the esets.cfg (***[pac]*** section). In this case, all exec functions of the LIBC are intercepted.

All opened, closed and executed files are scanned by the ESETS daemon for viruses. Based on the result of such scans, access to given files is denied or allowed.

5.3.2 Installation and configuration

The *libesets_pac.so* library module is installed using a standard installation mechanism of the preloaded libraries. One has just to define the environment variable *'LD_PRELOAD'* with the absolute path to the *libesets_pac.so* library. For more information, please refer to the *ld.so(8)* man page.

NOTE: It is important that the *'LD_PRELOAD'* environment variable is defined only for the network server daemon processes (ftp, Samba, etc.) that will be under control of the On-access scanner. Generally, preloading LIBC calls for all operating system processes is not recommended, as this can dramatically slow the performance of the system or even cause the system to hang. In this sense, the '/etc/ld.so.preload' file should not be used, nor should the 'LD_PRELOAD' environment variable be exported globally. Both would override all relevant LIBC calls, which could lead to system hang-up during initialization.

To ensure that only relevant file access calls within a given file system are intercepted, executable statements can be overridden using the following line:

```
LD_PRELOAD=/usr/lib/libesets_pac.so COMMAND COMMAND-ARGUMENTS
```

where 'COMMAND COMMAND-ARGUMENTS' is the original executable statement.

Review and edit the ***[global]*** and ***[pac]*** sections of the ESETS configuration file (esets.cfg). In order for the On-access scanner to function correctly, you must define the file system objects (i.e. directories and files) that are required to be under control of the preload library. This can be achieved by defining the parameters of the *'ctl_incl'* and *'ctl_excl'* options in the ***[pac]*** section of the ESETS configuration file. After making changes to the esets.cfg file, you can force the newly created configuration to be re-read by reloading the ESETS daemon.

5.3.3 Tips

In order to activate the On-access scanner immediately after file system start-up, the *'LD_PRELOAD'* environment variable must be defined within the appropriate network file server initialization script.

Example: Let's assume we want to have the On-access scanner to monitor all file system access events immediately after starting the Samba server. Within the Samba daemon initialization script (/etc/init.d/smb), we would replace the statement

```
daemon /usr/sbin/smbd $SMBDOPTIONS
```

with the following line:

```
LD_PRELOAD=/usr/lib/libesets_pac.so daemon /usr/sbin/smbd $SMBDOPTIONS
```

In this way, selected file system objects controlled by Samba will be scanned at system start-up.

6. Important ESET File Security mechanisms

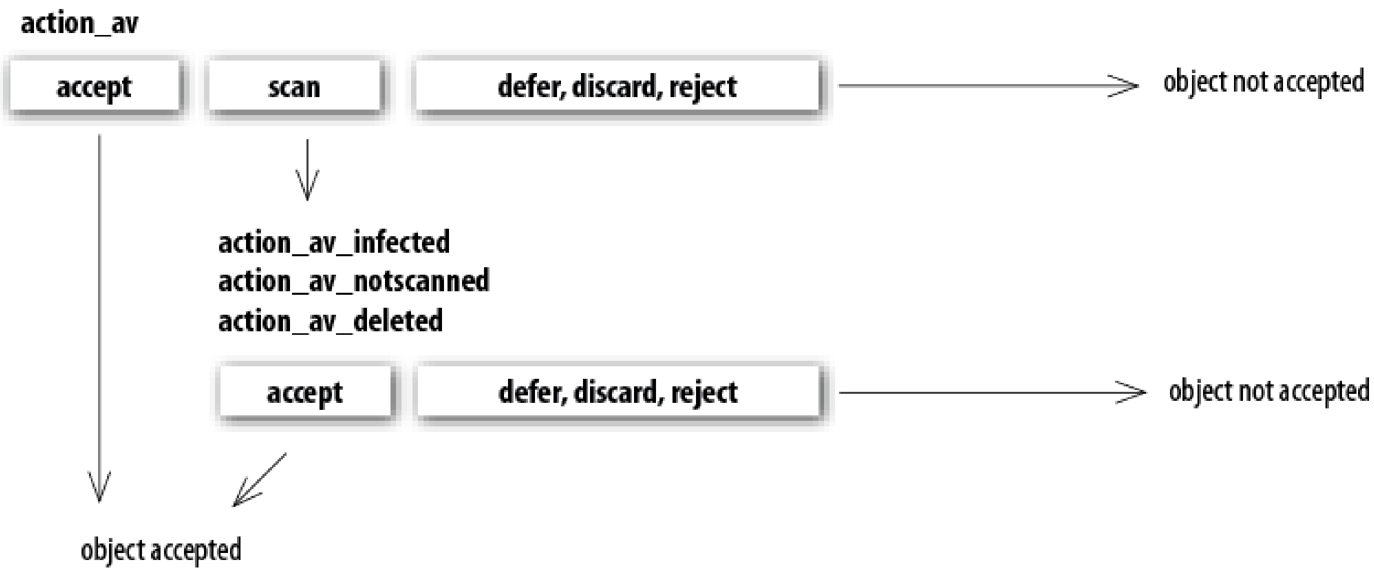
6.1 Handle Object Policy

The Handle Object Policy (see figure 6-1) mechanism provides filtering of scanned objects based on their status. This functionality is based on the following configuration options:

- action_av
- action_av_infected
- action_av_notscanned
- action_av_deleted

For detailed information on these options, please refer to the *esets.cfg(5)* man page.

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



Every object processed is first handled according to the configuration of the *'action_av'* option. If this option is set to *'accept'* (or *'defer'*, *'discard'*, *'reject'*) the object is accepted (or deferred, discarded, rejected). If the option is set to *'scan'* the object is scanned for virus infiltrations, and if the *'av_clean_mode'* option is set to *'yes'*, the object is also cleaned. In addition, the configuration options *'action_av_infected'*, *'action_av_notscanned'* and *'action_av_deleted'* are taken into account to further evaluate handling of the object. If an *'accept'* action has been taken as a result of these three action options, the object is accepted. Otherwise, the object is blocked.

6.2 User Specific Configuration

The purpose of the User Specific Configuration mechanism is to provide a higher degree of customization and functionality. It allows the sytem administrator to define ESETS antivirus scanner parameters based on the user who is accessing file system objects.

A detailed description of this functionality can be found in the *esets.cfg(5)* man page; in this section we will provide only a short example of a user-specific configuration.

In this example, the goal is to use the *esets_dac* module to control the ON_OPEN and ON_EXEC access events for an external disc mounted under the /home directory. The module can be configured in the ***[dac]*** section of the ESETS configuration file. See below:

```
[dac]
agent_enabled = yes
event_mask = "open"
ctl_incl = "/home"
action_av = "scan"
```

To specify scan settings for an individual user, the *'user_config'* parameter must specify the special configuration filename where the individual scanning rules will be stored. In the example shown here, the special configuration file is called *'esets_dac_spec.cfg'* and is located within the ESETS configuration directory (This directory is based on your operating system. Please see Terminology and abbreviations page).

```
[dac]
agent_enabled = yes
event_mask = "open"
ctl_incl = "/home"
action_av = "scan"
user_config = "esets_dac_spec.cfg"
```

Once the *'user_config'* file parameter is specified within the ***[dac]*** section, the *'esets_dac_spec.cfg'* file must be created in the ESETS configuration directory. Finally, add the desired scanning rules.

```
[username]
action_av = "reject"
```

At the top of the special section, enter the username to which the individual rules will be applied. This configuration will allow all other users attempting to access the file-system to be processed normally. i.e., all file system objects accessed by other users will be scanned for infiltrations, except for the user *'username'*, whose access will be rejected (blocked).

6.3 Samples Submission System

The Samples submission system is an intelligent *ThreatSense.Net* technology that collects infected objects which have been detected by advanced heuristics and delivers them to the samples submission system server. All virus samples collected by the sample submission system will be processed by the ESET virus laboratory and if necessary, added to the ESET virus signature database.

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. All information collected will be used only to analyze new threats and will not be used for any other purpose.

In order to activate the Samples Submission System, the samples submission system cache must be initialized. This can be achieved by enabling the *'samples_enabled'* option in the ***[global]*** section of the ESETS configuration file. To allow for the actual delivery of samples to the ESET virus laboratory servers, the parameter *'samples_send_period'* must also be specified in the same section.

In addition, users can choose to provide the ESET virus laboratory team with supplementary information using the *'samples_provider_mail'* and/or *'samples_provider_country'* configuration options. The information collected using these options will assist in providing the ESET team with an overview about a given infiltration which may be spreading over the Internet.

For more information on the Samples Submission System, refer to the *esets_daemon(8)* man page.

6.4 Web Interface

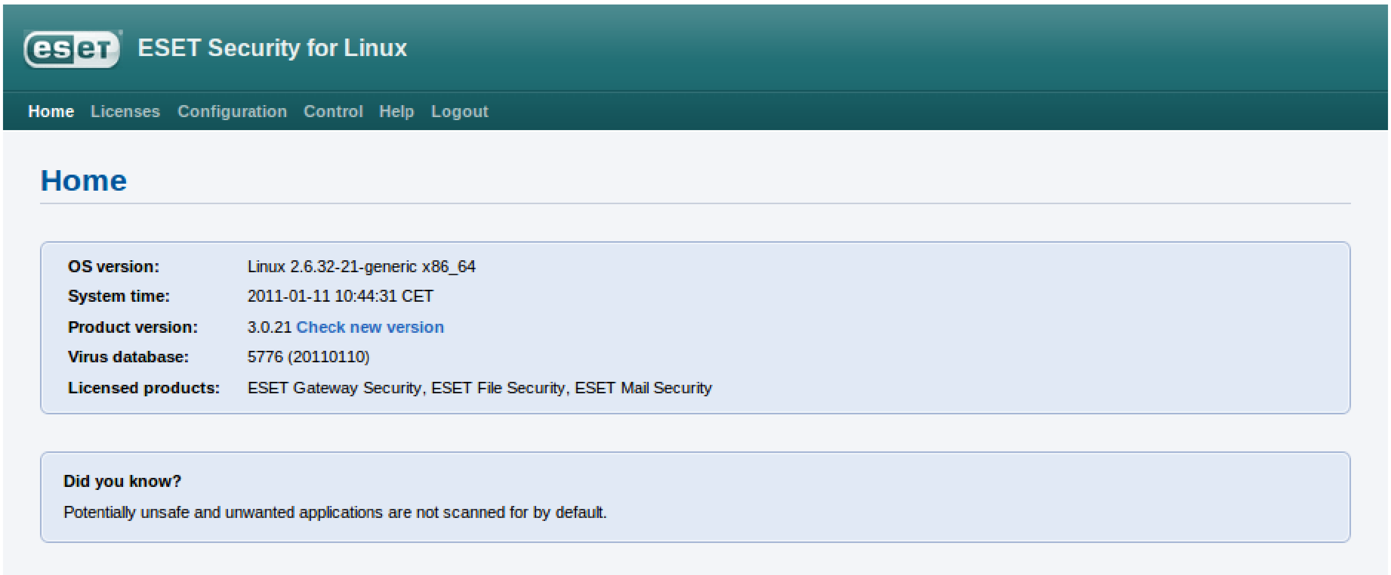
The Web Interface allows user-friendly configuration, administration and license management of ESET Security systems. This module is a separate agent and must be explicitly enabled. To quickly configure the *Web Interface*, set the following options in the ESETS configuration file and restart the ESETS daemon:

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

Replace the text in italics with your own values and direct your browser to *'https://address:port'* (note the https). Login with *'username/password'*. Basic usage instructions can be found on the help page and technical details about *esets_wwwi* can be found on the *esets_wwwi(1)* man page.

The web interface allows you to remotely access the ESETS daemon and deploy it easily. This powerful utility makes it easy to read and write configuration values.

Figure 6-1. ESET Security for Linux - Home screen.



Home

OS version: Linux 2.6.32-21-generic x86_64
System time: 2011-01-11 10:44:31 CET
Product version: 3.0.21 Check new version
Virus database: 5776 (20110110)
Licensed products: ESET Gateway Security, ESET File Security, ESET Mail Security

Did you know?
Potentially unsafe and unwanted applications are not scanned for by default.

The web interface window of ESET File Security is divided into two main sections. The primary window, that serves to display the contents of the selected menu option and the main menu. This horizontal bar on the top lets you navigate between the following main options:

- ***Home*** - provides basic system and ESET product information
- ***Licenses*** - is a license management utility, see the following chapter for mode details
- ***Configuration*** - you can change the ESET File Security system configuration here
- ***Control*** - allows you to run simple tasks and view global statistics about objects processed by esets_daemon
- ***Help*** - provides detailed usage instructions for the ESET File Security web interface
- ***Logout*** - use to end your current session

6.4.1 License management

You can upload a new license using the Web interface, as shown in Figure 6-2.

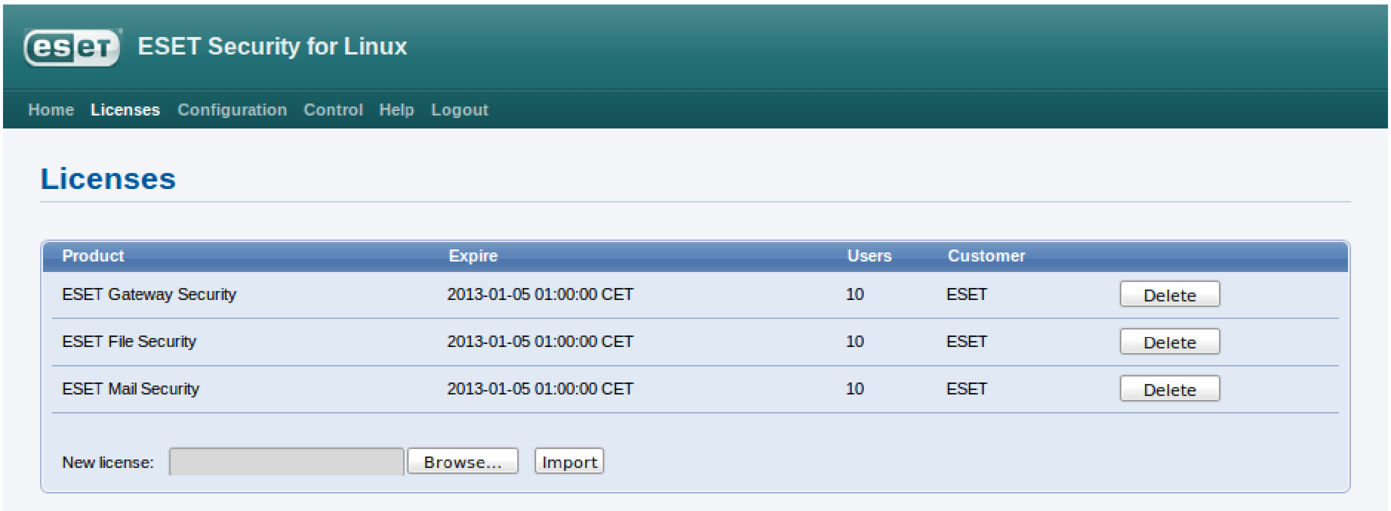
If you want to display licenses in the console, use the following command:

```
/usr/sbin/esets_lic --list
```

If you want to import new license files, use the following command:

```
/usr/sbin/esets_lic --import *.lic
```

Figure 6-2. ESET Licenses.



Licenses

Product	Expire	Users	Customer	
ESET Gateway Security	2013-01-05 01:00:00 CET	10	ESET	Delete
ESET File Security	2013-01-05 01:00:00 CET	10	ESET	Delete
ESET Mail Security	2013-01-05 01:00:00 CET	10	ESET	Delete

New license:

You can enable the license notification option in the ***Global*** section options. If enabled, this functionality will notify you 30 days prior to your license expiration.

6.4.2 On-Access scanner (DAC) configuration example

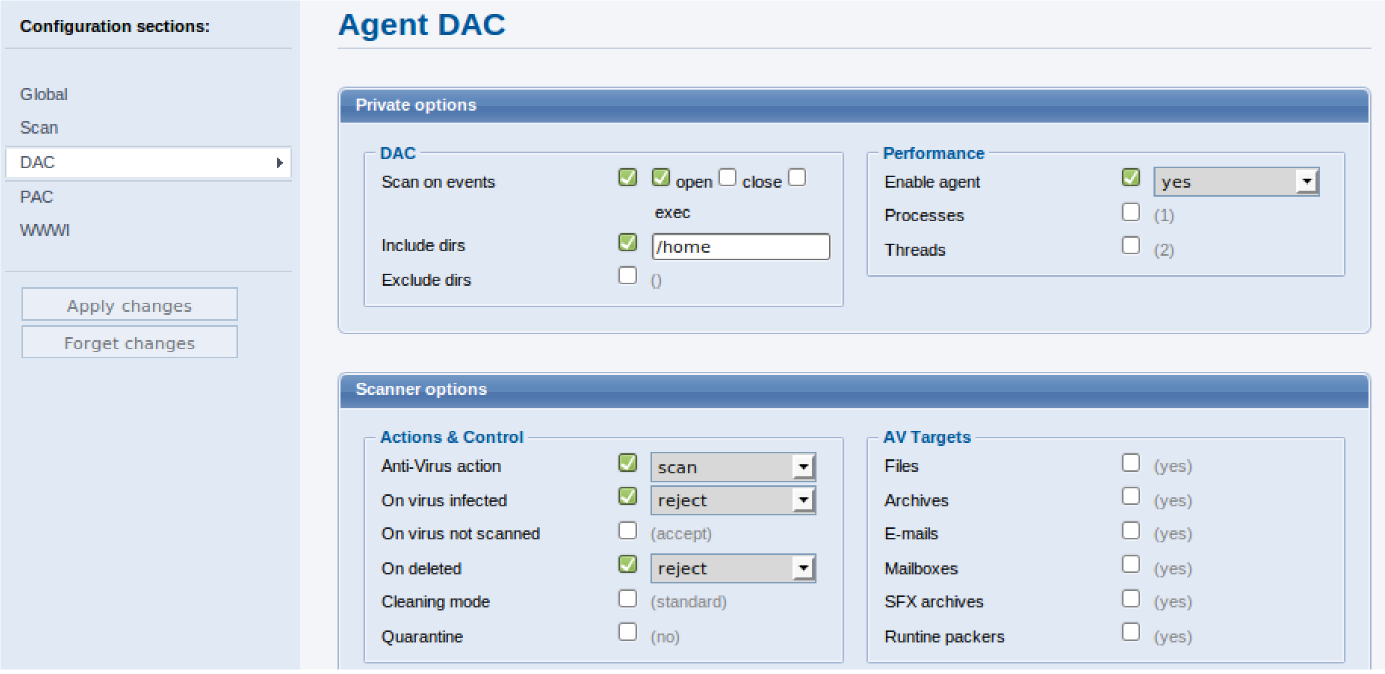
There are two ways you can to configure ESETS. In our example we will demonstrate how to use either of them to setup the DAC module, described in section 5.2. You can choose the option that best suits you.

- Using the ESETS configuration file:

```
[dac]
agent_enabled = yes
event_mask = "open"
ctl_incl = "/home"
action_av_deleted = "reject"
action_av = "scan"
action_av_infected = "reject"
```

- Using the web interface:

Figure 6-3. ESETS - Configuration > On-Access scanner.



Agent DAC

Private options

DAC
Scan on events ☑ ☑ open □ close □ exec
Include dirs ☑ /home
Exclude dirs □ ()

Performance
Enable agent ☑ yes
Processes □ (1)
Threads □ (2)

Scanner options

Actions & Control
Anti-Virus action ☑ scan
On virus infected ☑ reject
On virus not scanned □ (accept)
On deleted ☑ reject
Cleaning mode □ (standard)
Quarantine □ (no)

AV Targets
Files □ (yes)
Archives □ (yes)
E-mails □ (yes)
Mailboxes □ (yes)
SFX archives □ (yes)
Runtime packers □ (yes)

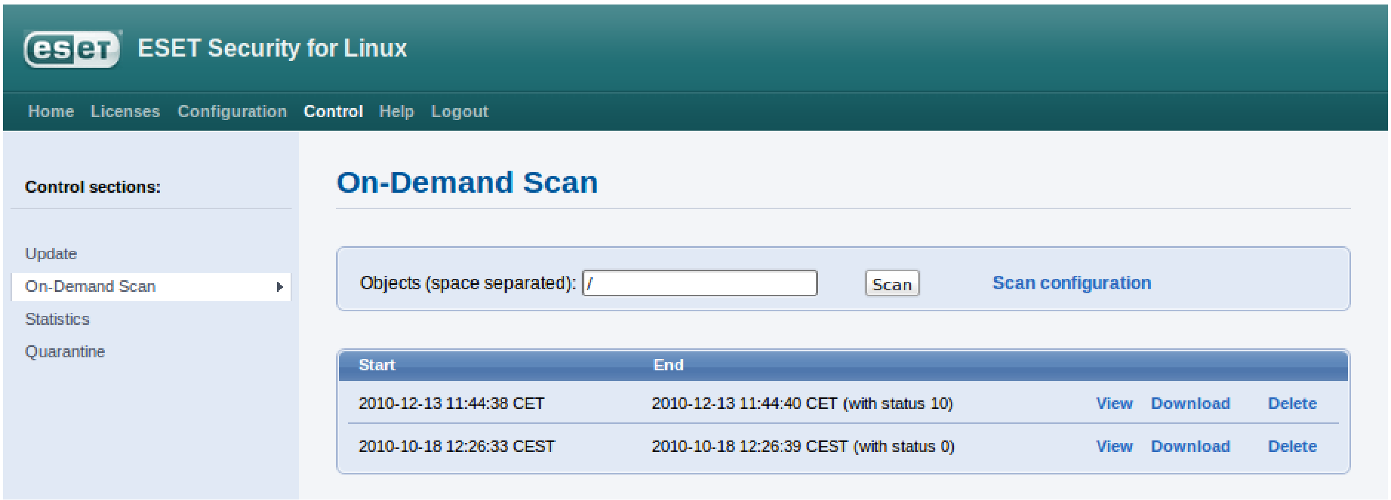
When changing settings in the web interface, always remember to save your configuration by the click ***Save changes***. To apply your new changes click the ***Apply changes*** button in the ***Configuration*** sections panel.

6.4.3 On-Demand scanner

This section comprises an example on how to run the On-Demand scanner to scan for viruses:

- Navigate to ***Control > On-Demand Scan***
- Enter the path to the directory you want to scan
- Execute the Command-line scanner by clicking the ***Scan*** button

Figure 6-4. ESETS - Control > On-Demand scanner.



On-Demand Scan

Objects (space separated): /

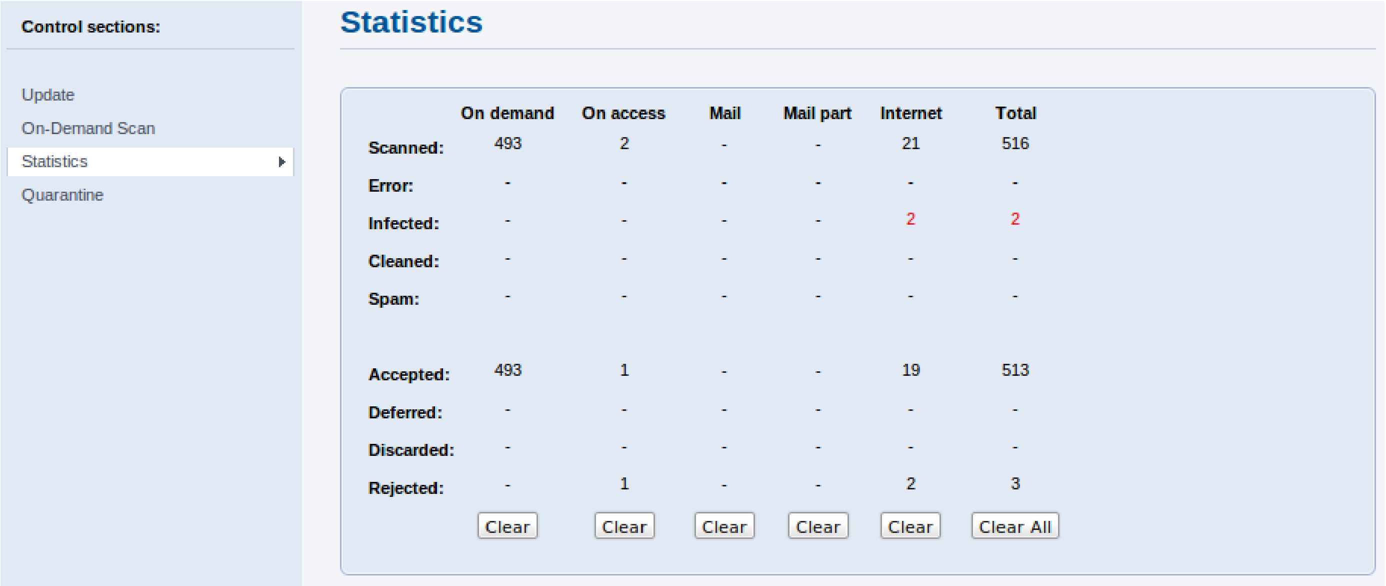
Start	End			
2010-12-13 11:44:38 CET	2010-12-13 11:44:40 CET (with status 10)	View	Download	Delete
2010-10-18 12:26:33 CEST	2010-10-18 12:26:39 CEST (with status 0)	View	Download	Delete

ESET Command-line scanner will automatically run in the background. To see the scanning progress, click the ***View*** link. A new browser window will open.

6.4.4 Statistics

You can view statistics for all of active ESETS agents here. ***Statistics*** summary refreshes every 10 seconds.

Figure 6-5. ESETS - Control > Statistics.



Statistics

	On demand	On access	Mail	Mail part	Internet	Total
Scanned:	493	2	-	-	21	516
Error:	-	-	-	-	-	-
Infected:	-	-	-	-	2	2
Cleaned:	-	-	-	-	-	-
Spam:	-	-	-	-	-	-
Accepted:	493	1	-	-	19	513
Deferred:	-	-	-	-	-	-
Discarded:	-	-	-	-	-	-
Rejected:	-	1	-	-	2	3
	Clear	Clear	Clear	Clear	Clear	Clear All

6.5 Remote Administration

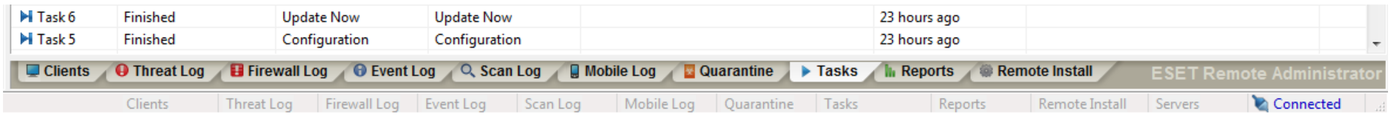
ESETS supports ESET Remote Administration for file security management in large computer networks. The ESETS Remote Administration Client is part of the main ESETS daemon and performs the following functions:

- Communicates with ERA Server and provides you with system information, configuration, protection statuses and several other features
- Allows client configurations to be viewed/modified using the ESET Configuration Editor and implemented with the help of configuration tasks
- Can perform *Update Now* tasks
- Performs On-demand scans as requested, and submits the resulting back to ERA Server ***Scan Log***
- Adds logs of notable scans performed by the ESETS daemon to ***Threat Log***
- Sends all non-debug messages to ***Event Log***

These functionalities are not supported:

- Firewall Log
- Remote Install

Figure 6-6. ERA Console tabs.



Task 6	Finished	Update Now	Update Now	23 hours ago
Task 5	Finished	Configuration	Configuration	23 hours ago

For more information, please read the ESET Remote Administrator manual. This manual is located on our web site at the following link:

http://www.eset.com/documentation

6.5.1 Remote Administration usage example

Before commencing any remote administration process ensure your system fulfills the three following prerequisites:

- Running ERA Server
- Running ERA Console
- Enable RA Client in the ESETS daemon. Ensure that firewall settings do not block traffic to ERA Server or vice versa.

To setup the basics, specify the address of your ERA Server in the *'racl_server_addr'* parameter first. If you are using a password to access the ERA Console password, you must edit the value of the *'racl_password'* parameter accordingly. Change the value of the 'racl_interval' parameter to adjust the frequency of connections to ERA Server (in minutes).

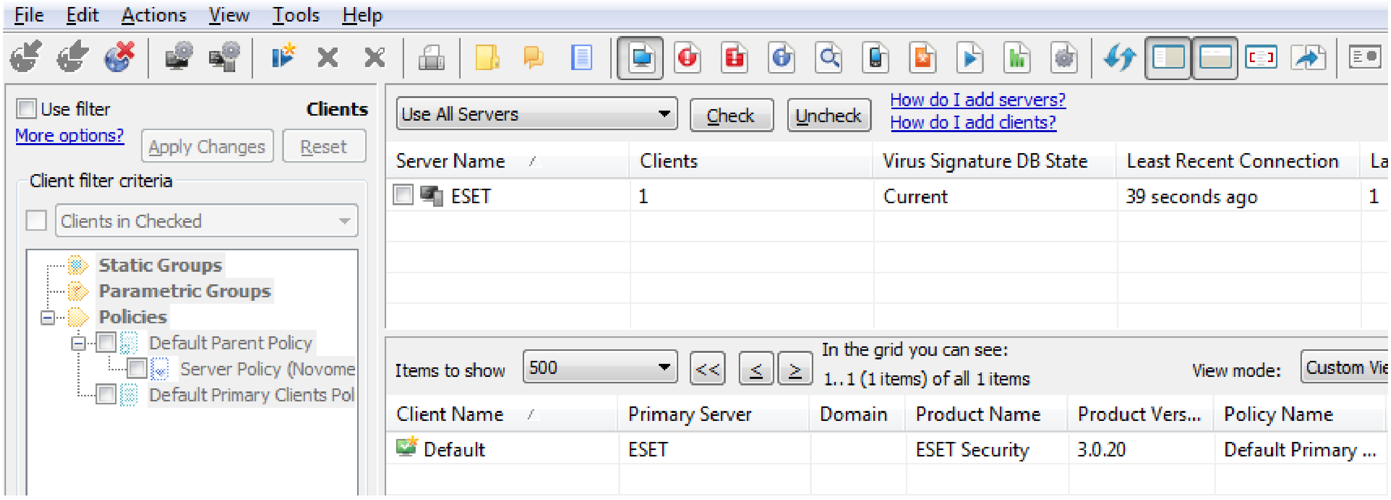
You can either use the web interface (see also previous chapter) to apply the new configuration, or you can adjust these parameters in the ***[global]*** section of the ESETS configuration file as follows:

```
racl_server_addr = "yourServerAddress"
racl_server_port = 2222
racl_password = "yourPassword"
racl_interval = 1
```

NOTE: All applicable ESET Remote Administration Client variables are listed on the *esets_daemon(8)* man page.

The ESETS daemon configuration will be reloaded and RACL will connect to ERA Server. You will be able to see a newly connected client in your ERA Console. Press the F5 button (or ***Menu > View > Refresh***) to manually refresh the list of connected clients.

Figure 6-7. ERA Console.



Use All Servers

Server Name	Clients	Virus Signature DB State	Least Recent Connection	La
ESET	1	Current	39 seconds ago	1

Items to show 500

In the grid you can see:
1..1 (1 items) of all 1 items

View mode: Custom Vie

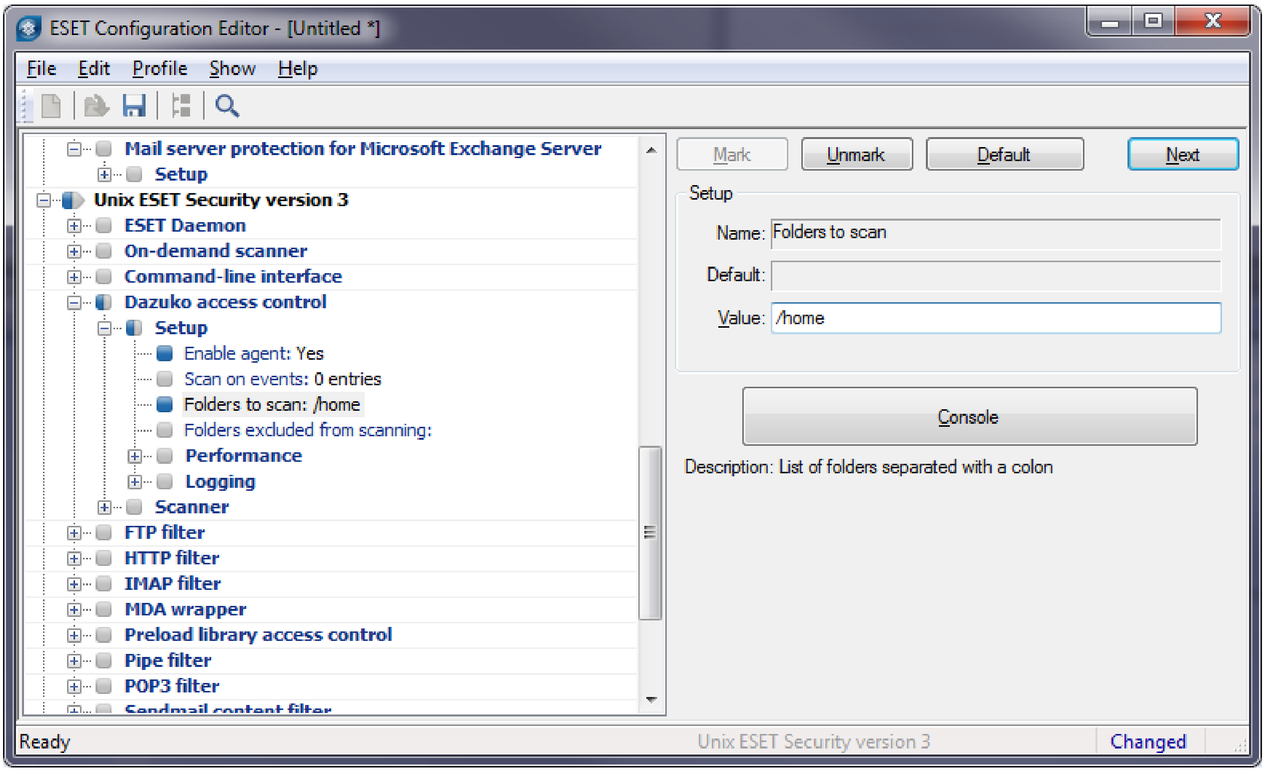
Client Name	Primary Server	Domain	Product Name	Product Vers...	Policy Name
Default	ESET		ESET Security	3.0.20	Default Primary ...

By using ERA Console you can create a configuration task to ESETS daemon from ERA Console:

- Right click the connected ***Client Name***
- Navigate to ***New Task > Configuration Task > Create...***
- Expand ***Unix ESET Security tree***

For an example of a configuration task by the DAC agent, see below:

Figure 6-8. ERA Configuration Editor.


ESET Configuration Editor - [Untitled *]
File Edit Profile Show Help
Mark
Unmark
Default
Next
Console

Mail server protection for Microsoft Exchange Server
Setup
Unix ESET Security version 3
ESET Daemon
On-demand scanner
Command-line interface
Dazuko access control
Setup
Enable agent: Yes
Scan on events: 0 entries
Folders to scan: /home
Folders excluded from scanning:
Performance
Logging
Scanner
FTP filter
HTTP filter
IMAP filter
MDA wrapper
Preload library access control
Pipe filter
POP3 filter
Sendmail content filter

Setup

Name: Folders to scan

Default:

Value: /home

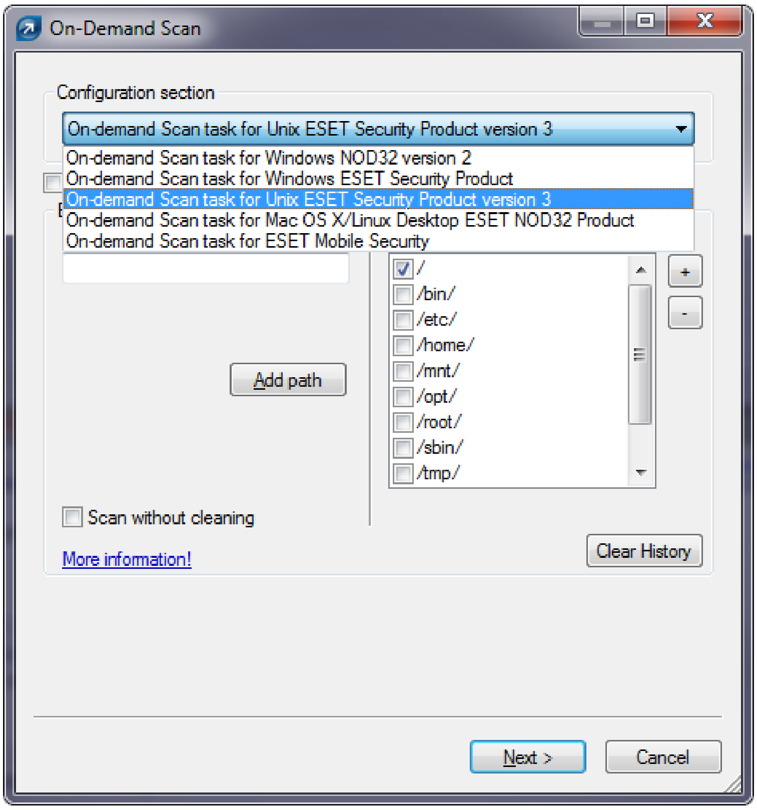
Description: List of folders separated with a colon

Ready | Unix ESET Security version 3 | Changed

The ***New Task*** context menu contains On-demand scanning options (enabled/disabled cleaning).

You can select the desired product, that you wish to set the task for, in the ***On-Demand Scan*** pop-up window in the ***Configuration Section*** drop-down menu. Make sure that you select the ***On-demand Scan task for Unix ESET Security Product*** option (i.e. the product that is installed on your target workstation).

Figure 6-9. ERA On-demand scan.



Configuration section

On-demand Scan task for Unix ESET Security Product version 3

- On-demand Scan task for Windows NOD32 version 2
- On-demand Scan task for Windows ESET Security Product
- On-demand Scan task for Unix ESET Security Product version 3
- On-demand Scan task for Mac OS X/Linux Desktop ESET NOD32 Product
- On-demand Scan task for ESET Mobile Security

☑ /
☐ /bin/
☐ /etc/
☐ /home/
☐ /mnt/
☐ /opt/
☐ /root/
☐ /sbin/
☐ /tmp/

☐ Scan without cleaning

6.6 Logging

ESETS provides system daemon logging via syslog. *Syslog* is a standard for logging program messages and can be used to log system events such as network and security events.

Messages refer to a facility:

```
auth, authpriv, daemon, cron, ftp, lpr, kern, mail, ..., local0, ..., local7
```

Messages are assigned a priority/level by the sender of the message:

```
Error, Warning, Summall, Summ, Partall, Part, Info, Debug
```

This section describes how to configure and read the logging output of syslog. The *'syslog_facility'* option (default value *'daemon'*) defines the syslog facility used for logging. To modify syslog settings edit the ESETS configuration file or use the web interface. Modify the value of the *'syslog_class'* parameter to change the logging class. We recommend you modify these settings only if you are familiar with syslog. For an example of syslog configuration see below:

```
syslog_facility = "daemon"
syslog_class = "error:warning:summall"
```

The name and location of the log file depend on your syslog installation and configuration (e.g. rsyslog, syslog-ng, etc.). Standard filenames for syslog output files are for example *'syslog'*, *'daemon.log'*, etc. To follow syslog activity, run one of the following commands from the console:

```
tail -f /var/log/syslog
tail -100 /var/log/syslog | less
cat /var/log/syslog | grep esets | less
```

If you enable ESET Remote Administration, ERA log entries older than given days by the option *'racl_logs_lifetime'* will be automatically deleted.

7. ESET Security system update

7.1 ESETS update utility

To maintain the effectiveness of ESET File Security, the virus signature database must be kept up to date. The esets_update utility has been developed for this purpose. See the *esets_update(8)* man page for details. To launch an update, the configuration options *'av_update_username'* and *'av_update_password'* must be defined in the ***[global]*** section of the ESETS configuration file. In the event that your server accesses the Internet via HTTP proxy, the additional configuration options *'proxy_addr'*, *'proxy_port'* must be defined. If access to the HTTP proxy requires a username and password, the *'proxy_username'* and *'proxy_password'* options must also be defined in this section. To initiate an update, enter the following command:

```
@SBINDIR@/esets_update
```

To provide the highest possible security for the end user, the ESET team continuously collects virus definitions from all over the world - new patterns are added to the virus signature database in very short intervals. For this reason, we recommend that updates be initiated on a regular basis. To specify the update frequency, the *'av_update_period'* option must be defined in the ***[global]*** section of the ESETS configuration file. The ESETS daemon must be up and running in order to successfully update the virus signature database.

7.2 ESETS update process description

The update process consists of two stages: First, the precompiled update modules are downloaded from the ESET server. If the option *'av_mirror_enabled'* is set to 'yes' in the ***[global]*** section of the ESETS configuration file, copies (or mirror) of these update modules are created in the following directory:

```
@BASEDIR@/mirror
```

If desired, the Mirror directory path can be redefined using the *'av_mirror_dir'* option in the ***[global]*** section of the ESETS configuration file. The newly created Mirror can then serve as a fully functional update server and can be used to create lower (child) Mirror servers. See section 7.3 for details.

The option *'av_mirror_pcu'* allows you to download Program Component Update (PCU) modules for Windows-based ESET security products. These modules can be mirrored from the ESET server.

NOTE: Once you set your username, password and license for ESET File Security to download PCU's for ESET NOD32 Antivirus / ESET Smart Security, please contact our Technical Support and request a change, that will enable your ESET File Security to download PCU's for our Windows-based products.

The second stage of the update process is the compilation of modules loadable by the ESET File Security scanner from those stored in the local mirror. Typically, the following ESETS loading modules are created: loader module (em000.dat), scanner module (em001.dat), virus signature database module (em002.dat), archives support module (em003.dat), advanced heuristics module (em004.dat), etc. The modules are created in the following directory:

```
@BASEDIR@
```

This is the directory where the ESETS daemon loads modules from and thus can be redefined using the *'base_dir'* option in the ***[global]*** section of the ESETS configuration file.

7.3 ESETS mirror http daemon

ESETS mirror http daemon is installed automatically with ESET File Security. The http mirror daemon starts if the option *'av_mirror_httpd_enabled'* in the ***[global]*** section of the ESETS configuration file is set to *'yes'* and the Mirror is enabled.

Options *'av_mirror_httpd_port'* and *'av_mirror_httpd_addr'* define the port (default 2221) and address (default: all local tcp addresses) where the http server listens.

The option *'av_mirror_httpd_auth_mode'* allows access authentication (default: none) to be changed to basic. The options *'av_mirror_httpd_username'* and *'av_mirror_httpd_password'* allow an administrator to define the login and password used to access the Mirror.

8. Let us know

Dear user, we hope this Guide has provided you with a thorough understanding of the requirements for ESET File Security installation, configuration and maintenance. However, our goal is to continually improve the quality and effectiveness of our documentation. If you feel that any sections in this Guide are unclear or incomplete, please let us know by contacting Customer Care:

http://www.eset.com/support

or use directly the support form:

http://www.eset.eu/support/form

We are dedicated to provide the highest level of support and look forward to helping you should you experience any problems concerning this product.

9. 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.