NAME

esets_imap − ESETS IMAP filter module.

SYNOPSIS

esets_imap [**OPTION** ...]

DESCRIPTION

The IMAP filter scans the communication between an imap client and server for viruses. **esets_imap** is not a full IMAP server. It receives requests from the IMAP client (so called MUA - Mail User Agent) and forwards them to the IMAP server. Once e-mail is requested by a IMAP client, it is first scanned for infiltrations before its delivery. The entire client-server e-mail communication is transparent, except for the e-mail itself, which is subject to virus scanning.

IMPORTANT: Note that **esets_imap** supports most common MUA programs, such as MS Outlook, Evolution, Mozilla Thunderbird and others. Note that there is restriction in **esets_imap** functionality when emails downloaded by Mozilla Thunderbird. An email in this case is requested and downloaded part by part and built directly by Mozilla Thunderbird. For this reason it is not possible to write proper information about the infiltrations found into the header and body of the email and thus the functionality is deactivated for this MUA.

SIGNALS AND DAEMON PROCESS STRUCTURE

In order to start **esets_imap**, the agent manager process must be enabled using parameter **agent_enabled** in section [imap] of the main ESETS configuration file. The manager process is responsible for the agent initialization and also for the maintenance of **num_proc** processes each running pool of **num_thrd** client/server sessions servicing threads. This version of ESETS implements 1 vs N threads per sessions model with asynchronous I/O operations, i.e. each thread is able to maintain multiple sessions (resp. multiple opened client/server connections) at a time. Note that each thread performs sessions until termination, resp. until hard termination of the agent processes.

TERM This signal causes termination of the agent processes, i.e. after **esets_imap** manager process obtained this signal it terminates all daughter threads and processes as soon as possible and terminates.

USR1 This signal causes so called hard termination of the agent processes, i.e. after **esets_imap** manager process obtained this signal it terminates immediately all daughter threads and processes and terminates.

CONFIGURATION FILE OPTIONS

The main ESETS configuration mechanism is assumed to be that using ESETS main configuration file. Note that most principles of this mechanism are described in the esets.cfg(5) manual page while this section contains only additional information related with the list of configuration file options valid for this particular module. Therefore we recommend user to become familiar with the above mentioned documentation prior reading this section.

ESETS MODULE PRIVATE OPTIONS

agent_enabled = *yes/no*

type: bool

default: no

Enables/disables operation of **esets_imap** daemon manager process.

num_proc = *value*

type: integer

default: *value* = 1

Defines number of **esets_imap** daemon processes.

num_thrd = *value*

type: integer

default: *value* = 2

Defines number of **esets_imap** daemon threads per process.

listen_addr = *addr*

type: string

default: no default

Listen on *addr* for client connections. *LA* can be a host name or IP address. When IP address is set to 0.0.0.0 then **esets_imap** listens on all available network interfaces.

listen_port = *value*

type: integer

default: no default

Listen on the specified TCP port number *value*.

server_addr = *addr*

type: string

default: no default

Connect to server on *addr* and retrieve e-mails.

server_port = *value*

type: integer

default: no default

Connect to TCP port number *value* on the server.

timeout_client = *value*

type: integer

default: *value* = 30

Timeout for the communication of **esets_imap** agent with imap client measured in seconds. The time limit starts to be measured since last message sent to the client. After no further message is sent during this period from the client the **esets_imap** closes connection with both the client and server.

ESETS MODULE COMMON OPTIONS

To get detailed description of ESETS module common options, please refer to the section **ESETS MODULE COMMON OPTIONS** of the **esets.cfg(5)** manual page.

ESETS SCANNER COMMON OPTIONS

To get detailed description of ESETS scanner common options, please refer to the section **ESETS SCANNER COMMON OPTIONS** of the **esets.cfg(5)** manual page.

USER SPECIFIC CONFIGURATION

The ESETS system implements possibility to define so called user specific configuration, i.e. relevant configuration parameters specific for client's and/or server's IP address can be defined.

As described in section **USER SPECIFIC CONFIGURATION** of esets.cfg(5) manual page the user

specific configuration is created when an appropriate special configuration section created within a special configuration file *path* referenced from this agent section (see main ESETS configuration file) by option **user_config =** *path*.

The header name of user specific section must be in general of the following format,

[s_addr/s_mask|c_addr/c_mask]

where 's_addr' is server's network IP address, 's_mask' is server's network mask (represented by plain number, specifying the number of 1's at the left side of the network mask), 'c_addr' is client's network IP address, 'c_mask' is client's network mask (represented by plain number, specifying the number of 1's at the left side of the network mask). Thus the part 's_addr/s_mask' represents server's network address range for which we would like to define special configuration and part 'c_addr/c_mask' represents client's network address range for which we would like to define special configuration.

Note that it is not mandatory to define both client's and server's parts of the header name. In this case the appropriate part not present within header name will be assumed to be not restricted. The following example shows definition of section header name compound only from the client's address range for which we would like to define special configuration.

[|192.168.1.0/24]

Please, note that thanks to '|' character present at the beginning of section header name, the main ESETS daemon knows that an appropriate IP address range represents the client's part of the section header name. In case you omit the character '|', the appropriate content of the section header name will be assumed to be its server's part as shown in an example below.

[192.168.1.0/24]

If user specific configuration defined, it will be used automatically, as this agent automatically provides main ESETS scanning daemon **esets_daemon** with the client and server IP addresses received from the header of transferred packet during the FTP communication.

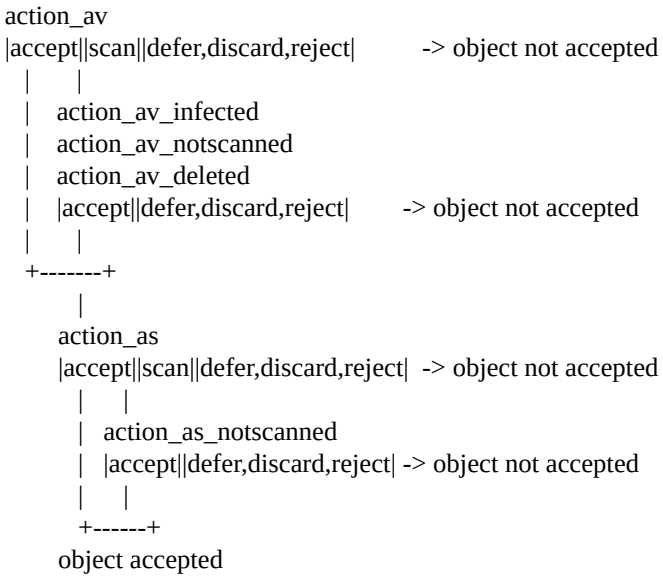
Once client's and/or server's IP address passed to the daemon, it is compared with section header names found in the special configuration file. The comparison is performed with all section header names consecutively in order as they are written within the file. The configuration appropriate to the first matched section is chosen. If no section header name matches the client's/server's IP address passed to the daemon, the configuration appropriate to the agent section from main ESETS configuration file is chosen. The section header name matching algorithm is as follows:

1. If no client's IP address passed to the daemon or no client's part of the section header name present, the algorithm returns match for this part of section header name. If client's IP address passed to the daemon, it is checked whether it matches network address range represented by client's part of the section header name.
2. Similarly if no server's IP address passed to the daemon or no server's part of the section header name present, the algorithm returns match for this part of section header name. If server's IP address passed to the daemon, it is checked whether it matches network address range represented by server's part of the section header name.

If both comparison steps described above return match the configuration appropriate to the section header name is chosen. On the other hand if at least one of the steps returns no match, an appropriate section is skipped.

HANDLE OBJECT POLICY

The Handle Object Policy (see figure below) is a mechanism that provides handling of the scanned e-mail messages depending on their scanning status. The mechanism implemented in this module is based on so called action configuration options **action_av**, **action_av_infected**, **action_av_notscanned**, **action_av_deleted**, **action_as**, **action_as_spam** and **action_as_notscanned**. To get description of these configuration options, see esets.cfg(5) manual page.



Every e-mail message processed by this module is first handled with respect to the setting of the configuration option **action_av**. Once the option is set to 'accept' (resp. 'defer', 'discard', 'reject') the object is accepted (resp. deferred, discarded, rejected). If the option is set to 'scan' the object is scanned (resp. also cleaned if requested by configuration option **av_clean_mode**) for virus infiltrations and set of action configuration options **action_av_infected**, **action_av_notscanned** and **action_av_deleted** 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 the object processed shall be scanned for spam.

Note that the e-mail message is scanned for spam only in case the configuration option **action_as** is set to 'scan'. In this case the action configuration options **action_as_spam** and **action_as_notscanned** is taken into account. If action 'accept' (resp. 'defer', 'discard', 'reject') has been taken as a result of the two above action options the object is accepted for further delivery (resp. the object is deferred, discarded or rejected).

You have probably noticed that each of the action configuration options discussed above accepts a variety of the values whose list can be found in esets.cfg(5) manual page. As also stated there the values listed are handled individually by every ESETS agent module. Thus to be consistent in the following we review the meaning of the values for this ESETS agent module.

accept Accept object on this level of Handle Object Policy, i.e. access to the object is allowed by the particular action configuration option.

scan Scan object for virus infiltrations (resp. for spam) and clean infected objects if requested by configuration option **av_clean_mode**.

defer Return temporary failure to sender.

discard
Accept object from sender, but drop it afterward.

reject Return permanent error to sender.

IMPORTANT: Note that any of 'defer', 'discard', 'reject' action can lead to accumulation of the e-mail messages on the server's disc until resources exhausted. Thus in the common it is not recommended to use of them. The action values are implemented by the module for special development purpose only.

LOGGING

Logging functionality of the ESETS agent modules has been developed to fine tune or to troubleshoot the agent module performance. Thus all the ESETS agent modules support only logging using syslogd daemon which logs system messages on *nix systems. To get more information on this topic please, refer to manual pages syslog(2), syslog.conf(5) and syslogd(8).

Regarding ESETS agent modules, this functionality can be invoked by setting ESETS module common

configuration option *syslog_facility* to value other than **none**. To get description of the introduced ESETS module common configuration options please, refer to esets.cfg(5) manual page.

Once the syslog logging enabled, the ESETS agent module messages are logged with one of the following syslog priorities:

LOG_ERR
Error messages concerned with the ESETS agent module performance are logged with this priority. Message logged with this priority usually means that error occurred during the ESETS agent module running and thus the module could not accomplish its operation or even the module process exited.

LOG_WARNING
Warning messages concerned with the ESETS agent module performance are logged or 'summary' messages concerned with action other than 'accept' taken as a consequence of object scanning status are logged with this priority.

LOG_NOTICE
The 'summary' messages concerned with action taken as a consequence of object scanning status are logged with this priority.

LOG_INFO
The common tasks are logged with this priority.

LOG_DEBUG
Debug information concerned with the ESETS agent module performance is logged with this priority.

COMMAND LINE OPTIONS

This section contains list of command line options valid for this module.

Note that present version of ESETS implements in general three types of command line options parameters:

Integer parameters accept integer values, e.g.:

--integer-parameter 26

For integer parameter it is also possible to append unit K (1K = 1024), M (1M=1024*1024), G (1=1024*1024*1024), e.g.:

--integer-parameter 4K

Boolean (logical) parameters do not accept any arguments. Their appearance in the command line automatically signal enabling of an appropriate functionality related with the parameter, e.g.:

--bool-parameter

Thus in order to provide negation of an appropriate logical parameter prescribe this parameter with --no, e.g.:

--no-bool-parameter

String parameters accept string values. They can be put into quotation marks, e.g.:

--string-parameter "string value"

Besides options on the command line, there are also commands. All ESETS programs accept at least these commands for displaying the version number and a simple help screen.

-v, --version

Print version information to stdout and exit.

-h, --help

Print help screen to stdout and exit.

REPORTING BUGS

In order to report bugs, please visit http://www.eset.com/support <URL:http://www.eset.com/support> or use directly the support form at http://www.eset.eu/support/form <URL:http://www.eset.eu/support/form>.

COPYRIGHT

SEE ALSO

esets_cgp(1) esets_cli(1) esets_dac(1) esets_ftp(1) esets_gwia(1) esets_http(1) esets_icap(1) esets_imap(1) esets_mda(1) esets_mird(1) libesets_pac.so(1) esets_pipe(1) esets_pop3(1) esets_smfi(1) esets_smtp(1) esets_ssfi.so(1) esets_wwwi(1) esets_zmfi(1) esets(5) esets.cfg(5) esets_daemon(8) esets_inst(8) esets_lic(8) esets_quar(8) esets_scan(8) esets_set(8) esets_update(8) eset_efs_userguide.pdf eset_egs_userguide.pdf eset_ems_userguide.pdf