

# User Guide

## **Vexira Antivirus for Mail Server**

for Linux/FreeBSD/Solaris

version 1.70





## TABLE OF CONTENTS

|  |           |
|--|-----------|
| <b>VEXIRA ANTIVIRUS FOR MAIL SERVER</b> .....  | <b>3</b>  |
| <b>System requirements</b> .....               | <b>4</b>  |
| <b>Program operating</b> .....                 | <b>5</b>  |
| <b>Installation</b> .....                      | <b>9</b>  |
| Installation on independent SMTP gateway ..... | 9         |
| Installation on mail server .....              | 9         |
| Installation steps .....                       | 10        |
| Manual installation .....                      | 13        |
| Uninstallation .....                           | 13        |
| <b>Registration</b> .....                      | <b>14</b> |
| <b>Safety of operation (Watchdog)</b> .....    | <b>14</b> |
| <b>Update database</b> .....                   | <b>15</b> |
| <b>Configuration</b> .....                     | <b>16</b> |
| Syntactical rules.....                         | 16        |
| [GLOBAL] General settings .....                | 18        |
| [Connector] Receiver/Sender settings .....     | 26        |
| [Target] Define target address .....           | 31        |
| [Message] Define messages .....                | 32        |
| Tokens .....                                   | 32        |
| [Template] Filter settings .....               | 35        |
| Virus filter .....                             | 38        |
| File filter .....                              | 40        |
| File gate .....                                | 41        |
| Field filter .....                             | 42        |
| Spam filter.....                               | 44        |
| Language filter .....                          | 51        |
| Black/White list, RBL .....                    | 52        |
| [Rules] Routing and rules .....                | 54        |
| LDAP support .....                             | 57        |
| LDAP authentication .....                      | 57        |
| Get settings from LDAP .....                   | 58        |
| Configuring Slapd-type LDAP server .....       | 60        |
| <b>Integration into mail servers</b> .....     | <b>61</b> |
| Sendmail.....                                  | 61        |
| Qmail .....                                    | 62        |
| Postfix .....                                  | 62        |
| Exim.....                                      | 63        |
| <b>END USER AGREEMENT</b> .....                | <b>65</b> |
| <b>CONTACT</b> .....                           | <b>67</b> |



## VEXIRA ANTIVIRUS FOR MAIL SERVER

The Vexira Antivirus for Mail Server package (hereinafter called VAMS) provides an effective protection for mailing systems on Linux, FreeBSD and Solaris platforms. The product checks and protects the mail traffic transmitted on the SMTP channel ensuring the secure (virus free) mailing. Apart from the anti-virus functions the program has routing, defense and filtering functions. As these functions are integrated they can be easily and rapidly configured and monitored from central computer. The program is very flexible and meets the requirements of all users.

### Main features of the product

-----

- Outstanding performance: parallel processing (multiprocess)
- Heuristic virus analysis to recognize unknown viruses
- Can be easily integrated into any mail system
- Incremental, automatic virus database update
- Spam filtering - statistical spam filtering with many evaluation methods, White/Black lists and RBL support
- LDAP support for almost all configuration options (SMTP authentication and LDAP-based smarthost support, individual filtering options for each user)
- Load balancing: the number of e-mail under processing, the number of inbound connections, the load on the mail filters can be set according to the total load on the server
- Flexible rule system to suit the company's needs
- WormBuster function - for blocking I-Worms instantly
- DirectScan - Enables mail scanning before accepting it - extended mail handling options on the receiver side (useful in case of using Postfix mail server)
- Ability to filter e-mails according to the language and script-type used in the e-mail
- Attachment's MD5 handling typescript script type
- Filtering based on mail header fields
- Filter out of password protected attachments
- Outstanding error handling to provide continuous functioning
- Customizable warning messages with tokens
- Informative statistics about mail traffic and events



## System requirements

Supported operating systems

-----

Linux (i386/amd64)  
FreeBSD 5.5, 6.0, 7.0 (i386/amd64)  
Solaris 9 (sparc), 10 (i386)

Minimal requirements

-----

Requirements for all the supported platforms:

- 256 MB free memory (512 MB recommended)
- 100 MB free hard disk space
- wget (for update)
- perl5 (for update)
- openLDAP 2.0.23 (for LDAP operations)

Requirements by platforms:

Linux, FreeBSD:

- Intel Pentium (or compatible) processor at 300 MHz
- Minimal required for Linux: GLIBC 2.2.5, kernel 2.2.1
- Minimal required Linux distributions: SuSE 8.0, RedHat 7.3, Debian 3.0 (woody), Mandrake 9.0, Slackware 8.1

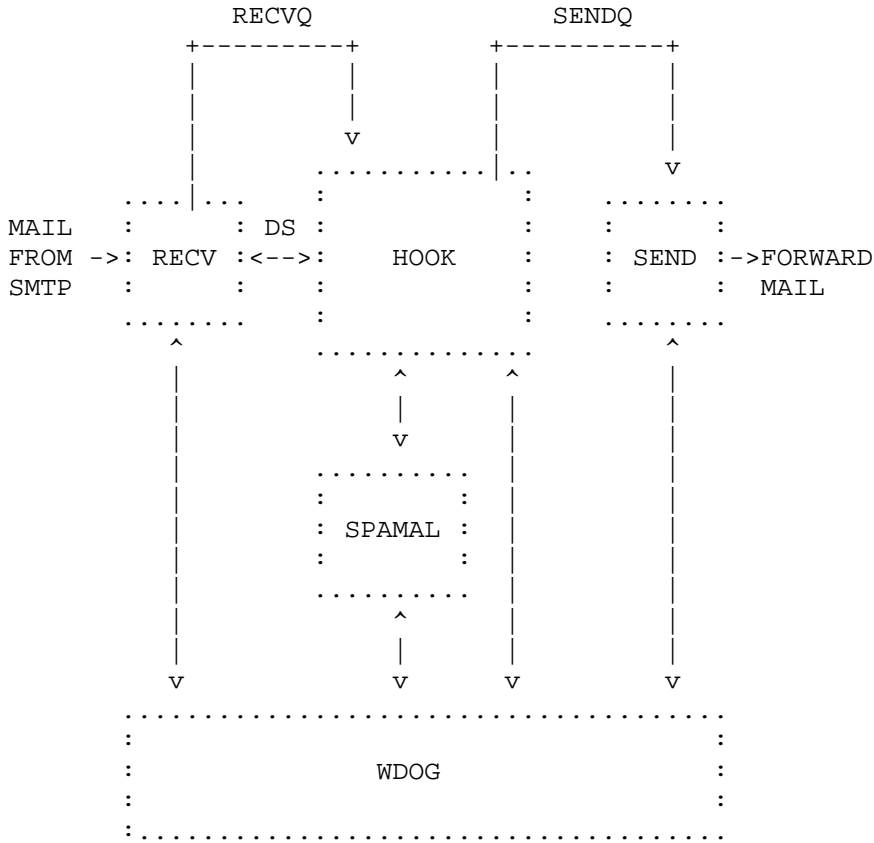
Solaris:

- Ultra Sparc IIe processor at 500 MHz



## Program operating

Architecture of VAMS:



- RECV: mail receiver module, prepares mails come from the smtp client, refuses or accepts mail based on the settings
- HOOK: main mail scanner module (hook), it scans the mail for viruses and filters it based on the settings (filefilter, fieldfilter, ...)
- SEND: mail forwarding module (sender), it delivers the mail to the recipient(s) or send notification message if needed
- RECVQ: wait queue for mails waiting for processing (receiver queue)
- SENDQ: wait queue for mails and notifications to be delivered (sender queue)
- DS: DirectScan operations, it is directly between the receiver and the hook if the DirectScan function is active
- SPAMAL: automatic spam learning module
- WDOG: control daemon, provides continuous operation of the system

Daemons found in the system  
-----

**vams [start|stop|hookstop|status|version|vdbreload|sdbreload|cfgreload|hookclean|senderclean|license|cfgcheck|language|list|-C <filename with path>]**  
Hook daemon, which processes the mails.

Parameters:

**start**  
Starts hook, receiver and sender daemons.

**stop**



## Vexira Antivirus for Mail Server

Stops receiver daemon and sends mails waiting for delivering then stops sender daemon, the and the watchdog.

hookstop

Stops VAMS hook only.

status

Lists processes and its PIDs belonging to the program.

version

Displays the version number of the program, virus database, virus scan engine and spam database (the last one's only if the program and the database have been loaded).

vdbreload

Reloads virus definition database.

sdbreload

Reloads spam database.

cfgreload

Reloads configuration file.

hookclean

Cleaning hook spool directory.

senderclean

Cleans sender spool directory.

license

Displays license information: username, registration key, expiring-date, status.

cfgcheck

Checks if the configuration file is correct.

languagelist

Returns supported languages/script-types available for the language filter.

start -C <filename with path>

VAMS will use an alternative configuration file set in the parameter.

Example:

```
start -C /usr/local/etc/.conf
```

Additional daemons:

**vasmtpd**

The connector receiver daemon.

**vasmtps**

The connector sender daemon.

**vawatchdog**

Watchdog daemon.

Starting daemons  
-----

Before starting daemons, the current smtp daemon (which connects to the number 25 port by default) must be stopped because it may occur binding error.



## Vexira Antivirus for Mail Server

Daemons can be started manually (individually). If the receiver and sender daemons are started, they will detect each other and create a connection. In this case, functions provided by the hook daemon are not active (file filtering, virus scan). This way a port-forward function can be formed with the help of the daemons if one of the direct-relaying or auto-relaying options are enabled. Routing is active in this case. If the hook daemon is started after this it will place itself between the other two daemons.

Attention!

If FreeBSD operating system is used, the following message can be occurred:

```
*** Engine Init Error.
```

```
Engine Init Error #: 65527
```

```
pid 4751 terminated abnormally! Exit code: 1.
```

To solve this problem, the system's kernel must be recompiled using the following option: option SHMMAXPGS=8192

Other executable files in the package

### **vashldstat [-r]**

Displays information updated continuously on VAMS's operation. All the data displayed on the screen have been generated since the last start of VAMS. Use the '-r' parameter to get simplified (raw) output, the elements will be listed in name:value form.

Explanation of displayed data:

**SMTP Receiver [PID]: x/y/z**

**Hook Scanner [PID]: x/y/z**

**Smtip Sender [PID]: x/y/z**

**PID - Daemon process ID**

x - Number of process at present (receiverek/hookok/sender).

y - Number of processes in the last 1 minute (receiverek/hookok/sender).

z - Maximum number of processes run at the same time (receiverek/hookok/sender).

### **Total connections**

Total number of connections to the VAMS.

### **Dropped connections**

Number of broken connections (for example: in case of there is no rule available for a mail).

### **Checked mails**

Total number of mails scanned by the VAMS.

### **Blocked mails**

Total number of mails blocked by the VAMS.

### **Received: x/y**

number of received mails (x) / amount of received data (y)

### **Sent: x/y**

number of sent mails (x) / amount of sent data (y)

### **System load average**

System load index number. It can display different value by platforms.

### **Queue status (R/S): x/y**

x - Number of mails in the receiver queue waiting for processing (A1).

y - Number of processed mails in the sender queue waiting for delivering (A2).

**Resend spool: x/y**

number of mails that wait for resend (x) / next resend process will be started in y second(s).

**Warning message sent**

Total number of warning messages sent.

**Virus found**

Number of detected viruses.

**IWorm found**

Number of detected IWorms.

**Infected mails**

Number of infected mails.

**Virus killed**

Number of viruses killed by the VAMS.

**Modified attachments**

Number of modified attachments.

**Deleted attachments**

Number of deleted attachments.

**File filtered**

Number of file filtered attachments.

**Spams found: x/y**

y - Total number of mails filtered by spam filter.

x - Number of mails marked as spam by the spam filter.

**Total spams found**

Total number of spams detected by the MailhShield.

**RBL****IP blacklist****Domain blacklist****Rcpt blacklist**

Number of mails sent by one of the black lists.

**5xx transmit.errors****4xx transmit.errors****Processing errors**

Number of mails produced one of the errors.

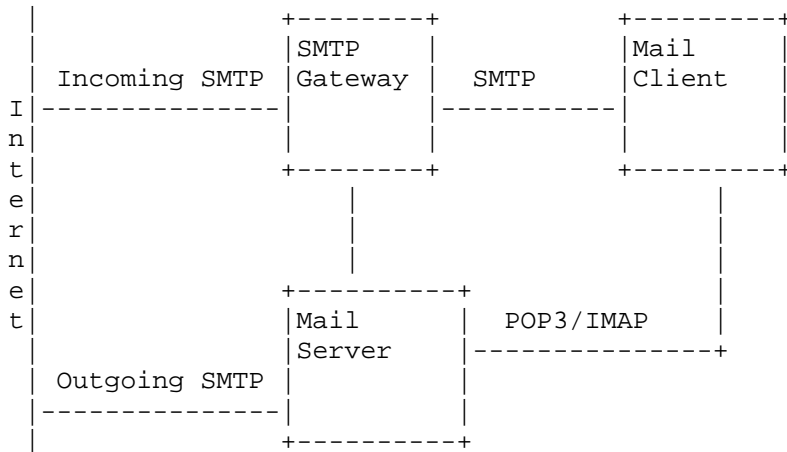


## Installation

The program (hereinafter called VAMS) can be installed on an independent SMTP gateway or on the same computer which executes the original mail server.

### Installation on independent SMTP gateway

You have to assure that all incoming and outgoing mails be first confronted with SMTP gateway.

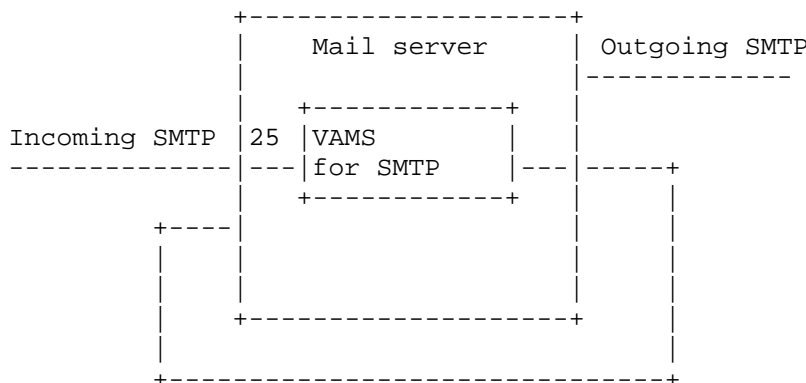


Install VAMS on SMTP gateway. This computer no needs other mailing system.

Your domain MX entry must be setup in such a way that incoming mails get to SMTP gateway. After processing it passes them to inside mail server. Users can download their mails from that.

In case of outgoing mail you have to setup mail clients to pass mails to the SMTP gateway which will transmit them to the mail server after processing.

### Installation on mail server



In this case you need to redirect mail server to other port. You can attain with this setting that mail server pays attention to your own port instead of standard port 25 of SMTP. VAMS receives mails through port 25 and passes them to the port scanned by mail server.



## Installation steps

Before installation, make sure that perl 5.005 or above and wget assistant program are installed on your system.

Before installing the new version of VAMS you should uninstall the previous version of that.

Unpack the package as follows:

```
#gzip -dc vams-<version>-<platform>.tgz|tar xvf -
```

In the course of unpacking there will be created a vams\_install folder which will contain all needed files. Enter this folder and run the install script:

```
#!/vams-install.pl
```

The script verifies if the uname file exists on the computer and the platform. Then there will be asked some questions:

In which directory do you want to install the binary files?

```
[/usr/bin]
```

You have to specify the name of the directory which stores VAMS, SMTPd, SMTPs binary files.

-----  
In which directory do you want to install the library files?

```
[/usr/lib]
```

You have to specify the name of the directory which stores library files. The program makes a /vams folder inside the specified directory and copies the virus definition database there.

-----  
In which directory do you want to install the documentation files?

```
[/opt/vams/doc]
```

You have to specify the name of the directory which stores description files.

-----  
In which directory do you want to install the manual files?

```
[/usr/share/man]
```

You have to specify the place of the manual files.

-----  
Which directory do you want to be the spool directory?

```
[/var/spool/vams]
```

You have to specify the name of the directory which stores mails under processing.

-----  
Specify the log directory name

```
[/var/log/vams]
```

You have to specify the directory of the log file.

-----  
Do you want to install the init script?

Press <y> key if you want to install init scripts.

-----  
Which directory contains the init scripts?

```
[/etc/init.d]
```

If you select install init scripts, you have to specify the name of the directory which stores initialize scripts.



## Vexira Antivirus for Mail Server

-----  
What is the directory that contains the init directories?  
[/etc]

You have to specify the place of the initialize directory.

-----  
Specify names which may appear on incoming mail messages  
Enter the domain names here separated with spaces.  
[]

You have to specify the part of domain/host of email addresses. Receiving mails working on exclusion concept which means that mails without having specified domain/host will be refused. For example if you want name@somewhere.com and name@mail.somewhere.com to be accepted then you have to specify both domain/host separated by space. In this case: somewhere.com mail.somewhere.com

-----  
Are there any networks of local machines you want to relay mail for?  
Enter them here, separated with spaces. You should use the standard address/length format (example: 194.222.242.0/24)  
If there are none, just press ENTER  
[]

You have to specify the network IP netmask. Essentially, you need to set your network netmask to allow outgoing mails. You have to determine the first IP address of your domain with the length of netmask. For example your domain is 192.168.10.1-192.168.10.255, netmask is 255.255.255.0 then you may register 192.168.10.1/24. You can specify several netmask separated by space.

-----  
Specify the IP address and port number of the SMTP server that will be used for forwarding scanned mail (example:192.168.1.2:2525)

You have to specify the IP address and port number of mail server which manages processed mails and warnings. If your server is redirected then you have to specify your own IP address or the localhost IP address (127.0.0.1) and the port number where the server is redirected to (for example: 127.0.0.1:2525).

-----  
Specify the log file name  
[vams.log]

You have to specify the name of the log file.

-----  
Specify the e-mail address of the administrator.  
[]

You have to specify the name of the network administrator who will receive warnings.

-----  
Specify the mail from field of administrator's email  
(example: admin@xxx.yy or Admin <admin@xxx.yy>)  
[]

You have to specify the from field of the mail sent by administrator in case any warnings.

-----  
Enter your registration user name  
[]  
Enter your registration key (example: WESAE-WCRVC-CSNEH)  
[]

You have to specify the registration name and key. (It is possible to use program without these data but see chapter 7. for more information)



## Vexira Antivirus for Mail Server

After these questions the program makes necessary directories and creates the configuration file in the /etc/vams directory according to the specified values. It generates initialize scripts and place them into an appropriate directory.



## Manual installation

Download and unpack the package according to the previous description..

Make the following directories:

```
/var/spool/vams  
/usr/lib/vams  
/etc/vams
```

Copy the files (vams vams-uninstall vashldstat vaSMTPd vaSMTPs) from the /bin directory (which can be found inside the install directory) into the /usr/bin directory.

Copy the libvengineso.1 file from the /lib directory (which can be found in the installation directory) into the /usr/lib directory and the copy the virus database files into the /usr/lib/vams directory. They can be found in the /usr/lib/vams directory.

Copy the directory from the /doc directory into the /usr/share/doc directory.

Copy the directory from the /man directory into the /usr/share/man directory.

Copy the vams.conf file from the /etc directory into the /etc/vams directory.

Copy the vams.sh script from the /etc directory into the system's directory which contains the initialization scripts. It may be different on different systems (e.g. /etc/init.d.)

Make the link in the initialization directory needed for the switching-over to the executing level. The place of the directories depends on your distribution.

For example:

```
/rc0.d  
ln -s /etc/init.d K20vams.sh  
/rc1.d  
ln -s /etc/init.d K20vams.sh  
/rc2.d  
ln -s /etc/init.d S20vams.sh  
/rc3.d  
ln -s /etc/init.d S20vams.sh  
/rc4.d  
ln -s /etc/init.d S20vams.sh  
/rc5.d  
ln -s /etc/init.d S20vams.sh  
/rc6.d  
ln -s /etc/init.d K20vams.sh
```

## Uninstallation

You have to execute the following program to uninstall the VAMS:

```
#vams-uninstall.pl
```



## **Registration**

The product can't be used without a valid registration key.

The program warns the user by sending a message into the log file and to the administrator once a day when the ending of the registration period is coming. After registration key had expired, the product works as before (without any restriction) until a program update (virus database updating is possible). After program updating, you need a new license (registration key) to use the program.

The registration key must be placed into the configuration file, see the description of the configuration settings for more.

## **Safety of operation (Watchdog)**

The built in Watchdog daemon provides continuous operating for the VAMS. It takes care of the other daemons of VAMS, checks their functions, maintains the continuous processing by preventing breakdowns of working processes.

The Watchdog controls the receiver, hook and sender daemons providing the continuously mail traffic, it handles the minor errors during operation, creates log. By the help of it the effects of the functional disorders can be reduced to the minimum level.

The control of the mail processing provides undisturbed traffic. In case of any errors the administrator can be notified and external programs are also executed specified in a command line option.

The Watchdog is able to provide the delivering even in case of repeated errors (you can set if mail will be copied into the failed spool or/and will be delivered without scanning).



## Update database

The update of the program's virus- and spam database can be performed manually or by the help of scripts that can be found in the package.

The product uses incremental virus database update mechanism to keep the virus database up-to-date. The advantage of this method is that the program doesn't need to download the whole virus database file every time (its size is several MBs) but usually only a small additional database package including the virus signatures processed and released recently. Using this mechanism, the download time is decreased to a minimal level so we can release additional virus database packages several times a day to improve the defense. Users can obtain protection against new malware without spending long time and generating considerable network load for the update. The protection is available almost immediately after the signatures of the newly discovered viruses have been processed in our virus lab.

### Manual update

Our virus database-set consist of several files, you need to update all the files from our FTP server from the following folder and copy them to the virus database folder:

`upd.vexira.com/pub12/vexira/vdb12/`

You can activate the new database by the "vams vdbreload" command.

The spam database file (vexira.sdb) is also available on our FTP server in compressed format:

`upd.vexira.com/pub12/vexira/sdb/sdb.tgz`

The downloaded file must be copied into the `/usr/lib/vams` directory as `vexira.sdb` (compressed version should have uncompressed before copying) and activated by the "vams sdbreload" command.

### Automatic update

We create scripts to automate the update process, they are in the `/usr/bin` directory (`vdbupdate.sh` and `vdbupdate_http.sh`).

Execute one of them, it is going to download the virus- and/or spam database, copies it/them into the correct directory and activates it/them. Updating will only be performed, if the database available in the server is newer than one on your computer. Otherwise the database will be left unchanged.

To execute the scripts, you should enter the `vdbupdate.sh` (through HTTP use the `vdbupdate_http.sh`) command. It is possible to use parameters, too:

`nosdb` - the spam database will not be updated  
`verbose` - display progress bar

Example:

```
vdbupdate.sh nosdb verbose
```

The spam database will not be updated, the progress bar will be displayed.

To run these scripts, you need `wget` program! By the help of `cron`, you can schedule the script executing to be performed by half an hours. Register into `/etc/crontab`:

```
0,30 * * * * root /usr/bin/vdbupdate.sh
```



## Configuration

The program working as a store-forward SMTP gateway. It has own daemon to receive incoming mails and pass them to Vexira Antivirus for Mail Server (hereinafter called VAMS) for processing. VAMS is scanning mails and files according to its configuration file and passes them to MTA together with possible warnings. MTA sends them to mail server to deliver.

The program hasn't full value MTA so it can't deliver mails without helping of an alternative mail server for the moment.

The configuration file of the program is `/etc/vams/vams.conf`.

## Syntactical rules

Syntactical rules used in the configuration file:

Space and tab characters are allowed to use free, but do not use special characters in the definition parts. In case of an error, the daemons display a message in the log file or on the standard output as follows:

**\*\*\* number. Error message**

If one of the sections contains errors, you can see the following error message:

**\*\*\* Config file parsing not successfully. Please check it!**

The daemons can only be started if the errors have been fixed.

### Sections, fields

-----  
The definition parts in the configuration file are divided into sections and fields. You can specify a section between [] signs. These include field names and their value(s). The fields in the section are related to each other. The name of the fields have a constant length, equal mark (=) is required to specify to assign a value. The value of the fields is determined in some cases. In case of error a warning message is displayed containing the type, location of the error and suggesting the fixing method.

### Comments

-----  
If a line contains the # character then any other characters placed after it will not be taken into consideration by the program (as these are comments).

### Joker characters

-----  
The documentation calls your attention to use joker characters where it is allowed. The effect of the joker characters (if there is no other disposition):  
The 'question mark' (?) character substitutes a piece of characters.  
The 'star' (\*) character substitutes any characters/character sequences.

### Assignment

-----  
Some of the options' assignments can be done in more lines. It is mentioned in the documentation where this feature is available.



**<option> = parameter(s)**

The specified parameter will be the value of the option.

**<option>+ = parameter(s)**

The specified parameter is appended to the previously specified value(s). The + character belongs to the option, it must be close to it.



## [GLOBAL] General settings

You can set the general parameters of the program in this section.

### **umask = 0077**

The VAMS will create its own files or directories with the specified permission. You can use this option the same way as the umask command that is available on the Unix systems.

Using the default umask setting set the -1 value to this option.

### **default-input = [IP\_address]:<port\_number>**

It determines where the SMTP daemon must receive the mails. If an IP address is specified then it receives only mails arriving at that IP address, otherwise all the IP address is scanned (0.0.0.0).

### **spooldir = <directory>**

Spool directory. The mails being currently processed are stored in this directory.

### **tempdir = <directory>**

Set a folder for temporarily created files.

### **connector = <label>**

Connector modul name. There is only one default connector section: "smtp".

### **libdir = <directory>**

Location of the library directory needed for the program. The virus database set must be available on this path.

Example:

```
libdir = usr/lib/vams
```

### **default-logfile = <file>**

Log file's name with path.

### **log-level =**

Values: -1-15 (log levels)

|         |    |
|---------|----|
| NONE    | -1 |
| EMERG   | 0  |
| ALERT   | 1  |
| CRIT    | 2  |
| ERR     | 3  |
| WARNING | 4  |
| NOTICE  | 5  |
| INFO    | 6  |
| DEBUG0  | 7  |
| DEBUG1  | 8  |
| DEBUG2  | 9  |
| DEBUG3  | 10 |
| DEBUG4  | 11 |
| DEBUG5  | 12 |
| DEBUG6  | 13 |
| DEBUG7  | 14 |
| DEBUG8  | 15 |

A statistic entry will be created from the INFO level into the log about each mail scanning as follows:

```
Mail summary: <mailfrom>|<rcpt to>[,<rcpt to2>,[<rcpt to3>]]|<ip_from>|<mailid>|<rbl>|<virusfilter>|<spamfilter>|<espfilter>|<zhfilter>|<filefilter>|<fieldfilter>
```

Example from a log file:



## Vexira Antivirus for Mail Server

Mail summary:

<sender@domain1.com>|<user1@domain2.com>, <user2@domain2.com>|192.168.2.1|A233457881|N|Y|N|N|N|N|

N and Y signs at the end of the entry: the mail was (Y) or wasn't (N) filtered by the current <filter>.

If the log level's size exceeds the 2 GB limit, new log events will not be recorded. Please use the "logrotate" daemon to maintain the log file.

### **syslog-facility =**

Syslog logging with facility and log level ('syslog-level') settings. The following facility values are available:

auth/authpriv/cron/daemon/ftp/kern/local0-7/lpr/mail/news/syslog/user/uucp

Default value: daemon

### **Important!**

Define the 'syslog-facility' before the 'syslog-level' option in the configuration file, otherwise the syslog facility will be set to the default.

### **syslog-level =**

Syslog level setting.

Level values are the same as 'log-level' option's. The NONE/-1 value disables the syslog.

### **found-virus-log = <yes/no>**

You can specify that if the incidents are logged in a special file.

### **virus-log-verb = <yes/no>**

It determines the fullness of the /var/log/FoundViruses.log .

Yes: also displays the sender and recipient(s) of the infected mail and the sender's IP address.

No: only displays the date of the virus incident, the name of the virus and the action.

### **virus-statistic-log = /var/log/vams/Statistic.log**

Set a log file (with path) to store the virus statistics.

### **found-viruses-log = /var/log/vams/Foundviruses.log**

Set a log file (with path) to register the virus incidents.

### **virus-top-list-log = /var/log/vams/Top-virus.log**

Set a log file (with path) for the daily virus top list.

### **port-banner = <yes/no>**

It determines that if the SMTP daemon returns the computer's full qualified domain name besides the number 220 and 421 SMTP answer.

### **banner-220 = <text>**

The login message of the SMTP daemon:

if port-banner set no the login answer is: "220 <text>"

if port-banner set yes the login answer is: "220 domain\_name <text>"

### **helo-name = <text>**

In case of (outgoing) communication to target, the VAMS will use the set name as host name in the 'HELO' SMTP command.

### **cfg-watch-timer = <second>**

Set the configuration check interval (seconds). When the time interval expires the program will check if the configuration file has been modified. The program reloads the configuration file if needed. After checking the interval starts again.

### **mail-archive = <yes/no>**



You can allow the VAMS saves all passing mails into the spool directory's archive subdirectory.

**error-patient = <number>**

Set the number of erroneous commands that should be accepted by the SMTP daemon before it disconnect the communication. Use the -1 value to disable this function.

**transparent-mode = <yes/no>**

Yes: warnings will not be sent in case of any actions (virus found, deleting mail or file).

**sum-warning-msg = <yes/no>**

If it is set 'yes', the warning messages are not sent by virus incidents to the administrator just a global message on the incidents.

**insert-attc-warn = <yes/no>**

If it is set 'yes', only warning message is sent instead of the file- or virus filtered attachments.

**language =**

The language of the program. Eng: English.

**attach-charset = <value>**

Character coding of replacement text file which is inserted when the source attachment is deleted. Default: iso8859-1

**block-partial = <yes/no>**

Certain mailing programs support sending big attachments in several parts. You can block these kind of attachments by the help of this option.

**block-partial-msg = <message label>**

Define the label of the notification message sent to the sender in case of blocking partial attachments.

**block-partial-smtp-reply = <error message>**

If both DirectScan function and 'block-partial' option are active, the specified error message will be passed to the smtp client in case of blocking.

Construct the error message in the way of it is described for the 'max-mail-per-connection-smtp-reply' option.

Default: 556 message (%msgid%) type: message/partial is not allowed

**attach-deleted-msg =**

**attach-deleted-msg+ =**

If you are not in transparent mode, system will substitute deleted attachment for an another file. You can specify text of the substituting file in case deleting infected file. You can use tokens in text.

**attach-filtered-msg =**

**attach-filtered-msg+ =**

If you are not in transparent mode system will substitute deleted attachment for an another file. You can specify the text of the substituted file in case deleting filtered file. You can use tokens in text.

**global-header-rewrite = <field:value>**

**global-header-rewrite+ = <field:value>**

**global-header-rewrite = +<field:value>**

Global header rewrite option. The value of the specified field in the header part will be modified according to the option's value. This option relates to all the mails. The value specified in +<field:value> form means inserting a new field (with its value) into the header. Tokens can be used in the option's value.

Example:

global-header-rewrite = Subject: %subject% checked by VAMS!



## **additional-sdb =**

Specifying additional spam database. Additional spam databases (with their path) must be enumerated separated by semicolon (;). The path is relative to the official spam database (vexira.sdb) location.

Example:

```
additional-sdb = custom.sdb; sth/sth.sdb; /sdb/new/new.sdb
```

## **archive-max-decomp-size =**

Default value: 0 (this time the program is using the virus scan engine's default value).

If this file size limit is exceeded during the decompression of an archive, the program stops this action and also the scan of the file and returns exploit virus found. (Option's value is in MByte).

If you change the default value of this option, archive-max-decomp-ratio value has also to be changed.

## **archive-max-decomp-ratio =**

Default value: 0 (this time the program is using the virus scan engine's default value).

Example value: 50 If the size of the decompressed file is 50 times (or more) greater than the compressed file's, the program will return exploit virus found.

Other explanation (option's value in percent):  $1/n*100$ , where n is the value

In the example:  $1/50*100 = 2\%$  so if the compression ratio is better than 2% the program will return exploit virus found.

If you change the default value of this option, archive-max-decomp-size value has also to be changed.

## **max-attachment-size = <max size>**

### **max-attachment-size-msg = <message label>**

If the size of one of the attachments including the mail is larger than the max size limit, the mail will be blocked and the specified message will be sent.

<max size>: it can be specified in bytes (no need to enter unit), in kilobytes (use 'k' or 'K') and in megabytes (use 'm' or 'M'), e.g.: 4096 or 10M or 100k. The 0 (zero) value allows unlimited size for attachments.

<message label>: Enter the label of the warning message. Tokens could be used in the message. In this case the %filename% token will contain the name of the invalid attachment.

### **max-attachment-size-smtp-reply = <error message>**

If DirectScan function is active, the specified error message will be passed to the smtp client in case of the above described size-limit exceeded.

Construct the error message in the way of it is described for the 'max-mail-per-connection-smtp-reply' option.

Default: 556 message (%msgid%) has exceeded maximum attachment size limit

## **max-attachment-count = <max attachments>**

### **max-attachment-count-msg = <message label>**

If the mail has more attachments attached than the specified, the mail will be blocked and the specified message will be sent.

<max attachments>: enter the number you wish to be the limit, the 0 (zero) value allows unlimited attachments

<message label>: Enter the label of the warning message. Tokens could be used in the message.

### **max-attachment-count-smtp-reply = <error message>**

If DirectScan function is active, the specified error message will be passed to the smtp client in case of the above described count-limit exceeded.

Construct the error message in the way of it is described for the 'max-mail-per-connection-smtp-reply' option.

Default: 556 message (%msgid%) has exceeded maximum attachment count limit

## **DirectScan**



-----  
DirectScan function is able to decide on the receiver-side if the system accepts or refuses the received mail so the sender will directly be informed about the result of the processing. Beside the reject-options which were available on the receiver side so far (blacklist, RBL, size-limit, ...), now also the result of the hook-side filters (virusfilter, filefilter, ...) is available to handle mail. User can specify a reply for each filter to be sent to the client in case of blocking. So, in case of a rejection triggered by a filter, it will be possible to use the tokens that were only available on the hook-side so far.

This function is useful in case of using Postfix mail server because the infected mails can be refused to the mail server before it disconnects from the antivirus system, so the mail's sender can be notified, too.

Because the mail-scan is performed directly by the hook, the processing speed of VAMS strongly depends on the system performance. So you are recommended to use this feature on high-performance machine in case of high intensity mail traffic.

The following options are available for setting DirectScan:

**directscan-enable = <yes/no>**

Enable/disable DirectScan function.

Default: no

**directscan-wait-queue = <number>**

The hook capacity to parallel processing is limited so the connections which were not accepted yet are in the queue waiting for processing. Set the maximum number of connections in the queue. Further connections will be refused.

If the value is 0, then 'recv-max-proc' option's value will be set for this option automatically.

Default: 0

**directscan-address = <unix path or ipv4 address>**

Hook address where the receiver connects to. Specify either unix socket (path) (e.g.: /var/tmp/vams-dscan.socket), or address (as address:port).

Default: /var/tmp/vams-dscan.socket

**directscan-partial-blocked-action = <accept/deny>**

In case of the mail has several recipients and different templates matches the mail with different actions (block and also forward) the mail will be handled by the setting defined in this option.

Available values:

accept: mail will be accepted and forwarded to all those who are not blocked

deny: mail will be refused and blocked (mail will not be forwarded)

Default: deny

The following time-out setting are in seconds:

**directscan-connect-timeout = <time-out>**

Time-out setting for establish connection between receiver and hook.

Default: 300

**directscan-recv-timeout = <time-out>**

Hook reply to the receiver time-out.

Default: 300

**directscan-send-timeout = <time-out>**

Socket send time-out setting.

Default: 60

## Load-balancing



-----  
**hook-max-proc =**

Maximum number of the hook processes. Increasing of this value can result in performance improvement but you shouldn't set too high value because of slowing down. Using greater value than the default is recommended in multi-processor systems.

**hook-load-avg =**

Load value in fraction (this setting needs to be multiplied with 10 to get the correct value in case of BSD systems).

**hook-reg-proc-max =**

Stopping control mechanism based on the load-avg value (specify as fraction). In case of reaching maximum process limit, this value will be the maximum process number in percent of max-proc in fraction.

**hook-load-reg-start =**

Starting control mechanism based on the load-avg value (specify as fraction).

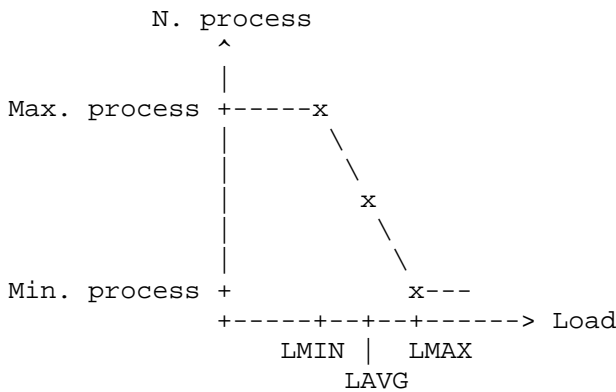
**hook-load-reg-end =**

Stopping control mechanism based on the load-avg value (specify as fraction).

Default settings:

- hook-max-proc = 6
- hook-load-avg = 6.0
- hook-load-reg-start = 0.5
- hook-load-reg-end = 1.25
- hook-reg-proc-max = 0.0 (see remark below for minimal number of processes)

Control mechanism:



Where:

- # Min. process = max-proc \* reg-proc-max
- # LMIN = load-avg \* load-reg-start
- # LMAX = load-avg \* load-reg-end

If the load reaches the LMIN value, then the VAMS reduces the maximum number of the processes according the load. If the load is greater than LMAX, then only 1 process will be executed.

Remarks:

- The program doesn't allow to set the process limit (in case of total down-control) less than 1
- If there is no hook-proc-max option set then this value will be equal with the value of proc-load-avg (the conversion of proc-load-avg to fraction is automatic)
- If rcv-max-proc option is not set then this value will be equal with the value of max-connections



## Warning messages

-----

**admin-mail = <e-mail\_address>**

The administrator's e-mail address. The admin messages is sent to this address.

**admin-mail-from = <e-mail\_address>**

The from field of the admin message generated by the VAMS in the name of the administrator (this setting appear in the sender field).

For example:

admin@xxx.yy

or

Admin admin@xxx.yy

**statistics-mail = <e-mail\_address>**

The statistics will be sent to the specified e-mail address.

The program generates the following warning messages by default:

# in case of error raising during virus database reload

(vdb-reload-error-msg)

# in case of any processes restarted by Watchdog

(proc-restart-msg)

# in case of any processes stopped by Watchdog where a repeated error occurred (proc-failure-msg)

You can define our own messages instead of these warning messages in the message section.

vdb-reload-error-msg = <label1;label2;...>

proc-restart-msg = <label1;label2;...>

proc-failure-msg = <label1;label2;...>

Tokens can be used in the messages.

## WatchDog settings

-----

**wdog-check-interval = 10**

Default checking interval (sec).

**wdog-mailproc-check-interval = 10**

Checking interval of Mailproc (sec).

**wdog-mailproc-save-failed = yes**

Yes: if mail processing error occurred, the mail will be saved to the failed spool.

**wdog-mailproc-forward-failed = yes**

Yes: If mail processing error occurred, the mail will be forwarded to the recipient(s).

**wdog-mailproc-max-proc-time = 60**

Available time for the hook process to work on an e-mail (sec).

**wdog-daemon-check-interval = 10**

Daemon checking interval (sec).

**wdog-daemon-restart-check-interval = 3600**

The max-restart is calculated within the specified interval (sec).

**wdog-daemon-max-restart = 5**



Within the check-interval the daemon can be restarted not more than the specified times (sec).

**wdog-mailproc-warning-msg = notify\_admin\_proc-failure1**

Label of the error message to be sent when the first mail processing error occurred.

**wdog-mailproc-failure-msg = notify\_admin\_proc-failure2**

Label of the error message to be sent when the second mail processing error occurred.

**wdog-error-command = <parancssor>**

Execute this external command if error occurred. Available tokens and their values:

Mailproc:

%errortype% "mailproc"

%error% "process does not exist" or "processing timeout"

%msgid% <message id (filename)>

%msgstate% "0" or "1" (0: first processing error, 1: second processing error)

Daemon:

%errortype% "daemon"

%error% "restart" or "failure"

%proccmdline% <cmdline argv[0]>

%procrestartcount% <restart count>

It is recommended to use tokens between quotes because if the string contains space characters it will be transmitted as one string.

### Registration

-----

**username = <registration\_name>**

**serialno = <registration\_key: ABCDE-ABCDE-ABCDE>**

The registration data must be specified.

Attention!

After modification of the registration key the VAMS must be restarted (vams stop, and start).



## [Connector] Receiver/Sender settings

**receiver = vasmtpd**

Receiver program's name (without path, because it must be in the path!)

**sender = vasmtps**

Sender program's name (without path, because it must be in the path!)

**auto-relaying = <yes/no>**

YES: The receiver will forward the mails to the sender even if the hook is not running.

**direct-relaying = <yes/no>**

YES: The receiver will forward the mails to the sender directly without any scanning.

**max-connections = nn**

Maximum number of the connections to the VAMS.

**max-hop = nn**

Maximum number of the 'Received' fields in the mail's header.

**insert-hop = <yes/no>**

YES: Allows to create your own 'Received' field.

**connect-timeout = nn**

Connect timeout in seconds.

**connect-backlog = nn**

Adjusts socket wait queue on the receiver side (backlog).  
Default: 20

**recv-timeout = nn**

Receiver socket read timeout (seconds).

**send-timeout = nn**

Sender socket write timeout (seconds).

**max-SMTP-size = 10M**

The maximum size of the received mail. If you specify the size, the 'k' means kilobyte, 'M' means megabyte (e.g.: 16k = 16 kilobyte, 4M = 4 Megabyte).

**max-SMTP-size-reply = <message>**

If the size limit set in the 'max-smtp-size' option exceeded, the error message specified in this setting will be returned.

**max-recipient = nnn**

The maximum number of the 'rcpt to' sent by the client.

**max-recipient-reply = <text>**

If the limit set in the 'max-recipient' option exceeded, the error message specified in this setting will be returned.

**strict-mailfrom = <yes/no>**

Enable (yes) or disable (no) using % @ ! characters in the SMTP 'mail from:' field. If it is enabled, some web pages searching for spam-forwarding MTAs may declare the server as open relay.

**deny-empty-mailfrom = <yes/no>**

Yes: VAMS doesn't accept empty character string in SMTP 'mail from:' field.  
No: it accepts empty character string in SMTP 'mail from:' field.



Some viruses spreading in e-mails use empty character string in the 'mail from:' field but usually the mail servers also use this value in the 'mail from:' field marking mails which can't be delivered.

**strict-rcptto = <yes/no>**

Enable (yes) or disable (no) using % @ ! characters in the SMTP 'rcpt to:' field.

**strict-relaying = <yes/no>**

This option controls the communication code returned during the denied relay.

YES: it returns error referring to denied-relay (558 Relaying denied)

NO: it returns just a warning message (250 Relaying denied)

Mail delivering to the recipient(s) will be blocked in both cases. It is recommended to set 'no' value to this option when the VAMS filters only the local network. In case of computers connected to the Internet it is recommended to set the 'yes' value to this option.

**recv-queue-full-warning-delay = nn**

After specified value (in sec.) system try to send a warning message to the administrator if receiver queue is full.

**smtp-cmd-read-char-mode = <yes/no>**

Receiving smtp commands per characters.

**recv-unavailable-type = smtp-421**

Available values: <smtp answer>, "none"

If VAMS is not able to receive the connection for any kind of reasons (for example more clients try to connect than allowed (this can be limited in the 'max-connection' option), here you can specify the SMTP answer.

In case of setting "none", the received connections wait until they are processed.

In case of SMTP-421 (Service not available, closing transmission channel SMTP error), a temporary error will be returned to the clients so they can try to connect later.

**real-ip-received-num = <N>**

VAMS uses the IP address of the specified 'Received: from' field placed in the header for filters whose need the mail's original IP address (e.g: RBL, IP black/white list). The 0 (zero) value means the connection IP. This option is useful in case of sequenced mail servers to determine the mail's original IP address. The value of the %realip% token will be equal to the IP specified here.

The anti-virus system is able to extract the IP address from RFC 821 or RFC 2821 compatible received field formats. In addition it also supports the Qmail field format.

The received field inserted by VAMS is not counted in this option so that the insert-hop setting does not have an effect on this function.

If the real-ip-received-num option's value is -1 then the program will compare the received fields with the ignore list's items (this list could be created in the [ip-ignore-list] section) to find which one is not listed in the [ip-ignore-list] section. Default items of the section:

```
[ip-ignore-list]
```

```
127.0.0.1/8
```

```
10.0.0.0/8
```

```
172.16.0.0/12
```

```
192.168.0.0/16
```

If the real-ip-received-num option's values is not -1, the [ip-ignore-list] will not be considered.

**rbl-action = <reject/stealth/modify>**

If the sender of the mail is on the list of the specified RBL's the following actions can be performed:

reject - rejecting mail (If the client is identified by client ip then the rejection will be performed right at the start of the communication, the mail will not be



received. If the client is identified by the received field then mail will be rejected at the end of the receiving process then it will be deleted. The client will be notified.)

stealth - Mail will be deleted after receiving, the sender will not be notified.

modify - Mail will be received, its header will be modified by 'rbl-header-rewrite' option.

**rbl-header-rewrite = <text>**  
**rbl-header-rewrite+ = <text>**  
**rbl-header-rewrite = +<text>**

If 'modify' value is set to the 'rbl-action' option then the text here specified will get into the header. Tokens can be used in the text value:

Example:

rbl-header-rewrite = Subject: Filtered by RBL!

**max-mail-per-connection = <mail limit>**  
**max-mail-per-connection-smtp-reply = <error message>**

Set the maximum number of mails per connections on the receiver side.

<mail limit>:

enter the number you wish to be the limit, the 0 (zero) value allows unlimited mails  
<error message>:

use the <error code> <message> form, build the error code as follows:

"[45][0-9][0-9]" e.g.: 452 Max mail/connection exceeded

**rcptto-rewrite = <none/ldap>**

Recipient alias resolving on the receiver side. Set the resolving method in this option. Available methods:

none - function disabled

ldap - resolving based on LDAP database

Default: none

**rcptto-rewrite-datasource = <datasource name>**

LDAP datasource name (consult the LDAP section for more).

**rcptto-rewrite-datasource-attribute = <attribute>**

Name of the attribute that contains the real e-mail address.

## Error-handling

-----

|                         |          |                 |
|-------------------------|----------|-----------------|
| send-io-error           | = action | Default: resend |
| send-smtp-4xx-error     | = action | Default: resend |
| send-smtp-5xx-error     | = action | Default: failed |
| send-smtp-other-error   | = action | Default: failed |
| send-mailloop           | = action | Default: failed |
| send-internal-error     | = action | Default: failed |
| send-max-retry-exceeded | = action | Default: failed |

Where:

Action = <resend/failed/delete>

# resend: moves mail to resend spool

# failed: moves mail to failed spool

# delete: deletes mail

In case of these errors warning messages can be sent by specifying the following options. You can determine message labels to represent messages in the set in the message section.

|                           |                       |
|---------------------------|-----------------------|
| send-io-error-msg         | = <label1;label2;...> |
| send-smtp-4xx-error-msg   | = <label1;label2;...> |
| send-smtp-5xx-error-msg   | = <label1;label2;...> |
| send-smtp-other-error-msg | = <label1;label2;...> |



```
send-mailloop-msg          = <label1;label2;...>
send-internal-error-msg    = <label1;label2;...>
send-max-retry-exceeded-msg = <label1;label2;...>
recv-queue-full-msg        = <label1;label2;...>
```

Tokens can be used in the messages.

The tokens above have the following meaning in case of send-smtp-4xx-error-msg, send-smtp-5xx-error-msg, send-smtp-other-error-msg messages:

```
%error%   - occurred comm. error's code and text
%lastcmd% - last smtp command
%msgid%   - message id
```

## Resend

-----

```
re-sending-X = Interval[Interval-modifier][,Retry]
Resending time-interval in case of unsuccessful sending.
```

Where:

```
X = 1-5
Interval          = decimal number (retry period)
Retry             = decimal number (number of retries)
Interval-modifier = [s,m,h,d,w] (sec.,min.,hour,day,week)
```

Default values (if they are not appeared in configuration file):

```
re-sending-1 = 5m, 12 ( 5 min x 12 = 1 hour)
re-sending-2 = 20m, 6 ( 20 min x 6 = 2 hour)
re-sending-3 = 4h, 5 ( 4 hours x 5 = 20 hours)
re-sending-4 = 12h, 2 ( 12 hours x 2 = 1 day)
re-sending-5 = 1d, 1 ( 1 day x 1 = 1 day)
```

END keyword can be used to shorten the list. E.g.:

```
re-sending-1 = 5m, 12
re-sending-2 = END
```

re-sending-X (where X>5) will not be processed

## Load-balancing

-----

```
recv-max-proc =
Maximum number of the receiver processes.
```

```
recv-load-avg =
Load value in fraction (this setting needs to be multiplied with 10 to get the correct value in case of BSD systems).
```

```
recv-reg-proc-max =
In case of maximum down-control the maximum number of the processes expressed in percent of max-proc in fraction.
```

```
recv-load-reg-start =
Beginning of the control in relation to load-avg, in fraction.
```

```
recv-load-reg-end =
Ending of the control in relation to load-avg, in fraction.
```

```
Default settings:
recv-max-proc = 8
recv-load-avg = 2.0
recv-load-reg-start = 0.5
```



## Vexira Antivirus for Mail Server

```
recv-load-reg-end = 1.25  
recv-reg-proc-max = 0.0
```



## [Target] Define target address

Target address can be specified below the [target <label>] section of the configuration file.

```
[target <label>]
server = <host_name/IP_address>
port   = <port_number>
bydns  = <first/last/none>
```

### <label>

The name of the target, this name is used in the rules

### <host\_name/IP\_address>

Host name or IP address of the smarthost.

### <port\_number>

The TCP port number scanned by the smarthost.

### bydns = <first/last/none>

Resolve target by dns mx record.

none

This function disabled (default).

first

First it tries to determine target host by mx record. If it fails then tries to determine IP by the specified smarthost.

last

First it tries to determine target host by smarthost. If it fails then tries to determine IP by dns mx record.

It is possible to get targets from LDAP. To use this function, you should make other setting needed for LDAP access. If several addresses are given in one line separated by | (pipe) then they will be used sequentially. The order also means priority. See the LDAP chapter for more information on setting.

### [target example]

```
server = LDAP(ds_smarthost, smarthostIP)|127.0.0.1
port   = LDAP(ds_smarthost, smarthostIP)|25025
```

One default target must be existed.



## [Message] Define messages

You can specify messages to the administrator in the 'Admin Notify Section' or to the sender and recipient(s) in the 'User Notify Section' inside the configuration file.

You can determine a message as follows:

```
[message <label>]

type      = <all/killed/deleted/blocked/forwarded>
language  = <eng>
address   = <e-mail_address>
[body]
Subject:  <mail_subject>

<Message>

[end]
```

```
type = <all/killed/deleted/blocked/forwarded>
Type of the message:
all          # in case of all incident
kill         # in case of killable virus killing
deleted      # in case of deleting attachment
blocked      # in case of deleting mail
forwarded    # in case of forwarding mail
If type is not defined, the default is 'all'.
```

```
language = <eng>
Language of the message:
eng          # English
```

```
address = <e-mail_address>
<%sender%/%recipient%/%admin%/any_e-mail_address>
The recipient of the message.
```

## Tokens

Tokens and their availability:

H: in hook  
S: in sender  
R: in receiver  
W: in watchdog

|              | H | S | R | W |                                 |
|--------------|---|---|---|---|---------------------------------|
| %fullresult% |   |   |   |   |                                 |
| %admin%      | x | x | x | x | system admin's e-mail address   |
| %version%    | x | x | x | x | version of the program          |
| %hostname%   | x | x | x | x | hostname                        |
| %msgid%      | x | x |   | x | received message's identifier   |
| %msgstate%   |   |   |   |   | -> see below                    |
| %peerip%     | x | x | x |   | client's ip like '11.22.33.44'  |
| %realip%     | x | x | x |   | -> see below                    |
| %sender%     | x | x |   |   | sender's address                |
| %recipient%  | x | x |   |   | recipient's address between < > |
| %rcptdomain% | x | x |   |   | recipient address' domain part  |



|                                 |   |   |                                    |
|---------------------------------|---|---|------------------------------------|
| <code>%rcptuser%</code>         | x | x | recipient address' user part       |
| <code>%action%</code>           | x | x | performed action                   |
| <code>%error%</code>            | x | x | error code in case of error        |
| <code>%tmpname%</code>          | x |   | attached file's temporary name     |
| <code>%vdbversion%</code>       | x |   | version of virus database          |
| <code>%vdbdate%</code>          | x |   | date of the virus database         |
| <code>%fullresult%</code>       | x |   | summarized report                  |
| <code>%filename%</code>         | x |   | attachment file's name             |
| <code>%virusname%</code>        | x |   | name of the found virus            |
| <code>%subject%</code>          | x |   | mail's subject                     |
| <code>%from%</code>             | x |   | mail's from field                  |
| <code>%to%</code>               | x |   | mail's to field                    |
| <code>%errortype%</code>        | x | x | explaining text for error occurred |
| <code>%lastcmd%</code>          | x |   | last command (hook, send, rcv)     |
| <code>%target%</code>           | x |   | target identifier                  |
| <code>%vdbloaderror%</code>     | x |   | vdb reload error                   |
| <code>%proccmdline%</code>      |   | x | restarting cmdline of daemon       |
| <code>%procrestartcount%</code> |   | x | number of daemon's restarting      |
| <code>%shieldbuild%</code>      | x |   | VAMS's build number                |
| <code>%sdbversion%</code>       | x |   | sdb (spam database) version        |
| <code>%template%</code>         | x |   | name of the current template       |
| <code>%shieldversion%</code>    | x |   | VAMS's version                     |
| <code>%fieldfilter%</code>      | x |   | Decision of field filter           |
| <code>%filefilter%</code>       | x |   | Decision of file filter            |
| <code>%filegate%</code>         | x |   | Decision of file gate              |
| <code>%virusfilter%</code>      | x |   | Decision of virus filter           |
| <code>%spamfilter%</code>       | x |   | Decision of spam filter            |
| <code>%languagefilter%</code>   | x |   | Decision of the language filter    |

## More information

### `%recipient%` vs `%rcptuser%@%rcptdomain%`

The `%recipient%` token result e-mail address between < and > signs so that the null value could also be displayed (<>).

If you need the 'real' e-mail address you can use the `%rcptuser%` and `%rcptdomain%` tokens to build this as follows: `%rcptuser%@%rcptdomain%`

### `%fullresult%`

Makes a summarized report on virus found, displaying every incident in separate line as follows:

```
virus_name      attachment_name  action
```

You can use it in case of `sum-warning-msg` set 'yes'. The system will send only a summarized mail not by incidents.

### `%msgstate%`

Its value can be '0' or '1'. '0' means the first processing error, '1' means the second error handled by WatchDog.

### `%realip%`

Its value is an IP address determined by the 'rbl-received-num' option. If this is not available, its value is equal to `%peerip%`.

### `%fieldfilter%`

Decision of field filter: 'N/A', 'NONE', 'MATCHES'

### `%filefilter%`

Decision of file filter: 'N/A', 'NONE', 'MATCHES'

### `%filegate%`

Decision of file gate: 'N/A', 'NONE', 'MATCHES'

### `%virusfilter%`

Decision of virus filter: 'N/A', 'NONE', 'INFECTED'

### `%spamfilter%`



Decision of spam filter: 'N/A', 'NONE', '0', '1', ...

**%languagefilter%**

Decision of language filter: 'N/A', 'UNRECOGNIZED', found script-type/language

Explanation:

N/A: filter was not applied or may not be applied

NONE: There was no found.

MATCHES: Found.

INFECTED: Infected attachment found.

0, 1, ..: Spam filter level on which spam was found.

UNRECOGNIZED: unrecognized



## [Template] Filter settings

There are several filter types available to check mails:

Field filter (fieldfilter)  
Virus filter (virusfilter)  
File gate (filegate)  
File filter (filefilter)  
Language/script-type filter (languagefilter)  
Spam filter (spamfilter)

The filters will always be performed according to this order. Disabled filters will not be performed.

Filter types can be organized in groups (so called templates) so that related filter types can be handled easily.

You can create a new filter section as follows:

```
[template <label>] <filter_type>
```

### <label>

Any unbroken character sequence, that identifies the template. The specified <filter\_type> will belong to the <label> template.

### <filter\_type>

Type of the filter. Specify one of the filters from the list mentioned before.

## Templates' inheritance

-----

Templates can be originated from any other templates, so in such a case all the options and settings of the original template will be transmitted to the new one. The inherited options and settings are allowed to modify by redefining them in the child template.

Default template (named 'default') must always be existed in the configuration file.

Create inherited template:

Use the following method to create inherited template with its all filters:

```
[template <parent_template> -> <child_template>]
```

Example:

```
[template original -> created]
```

If only one filter should be inherited from a template, use the following:

```
[template parent_template -> child_template] filter_name
```

Example:

```
[template original -> created] virusfilter
```

With the help of this method, you can simply change the settings of a filter in the selected template.

If a template is originated from a multi-level one, it will include all the levels of the original template. In a multi-level filter, the value of the settings on each new level will be originated from the previous level.



The following options are interpreted by each filter type

-----

|                     |                               |
|---------------------|-------------------------------|
| enable              | = <yes/no>                    |
| action              | = <forward/block/reroute>     |
| messages            | = <label1;label2;...>         |
| quarantine-path     | = <quarantine_directory_path> |
| quarantine-copy     | = <yes/no>                    |
| header-modify       | = <yes/no>                    |
| rewrite             | = <field:text>                |
| rewrite+            | = <field:text>                |
| rewrite             | = +<field:text>               |
| forward-to          | = <e-mail_address>            |
| forward-copy        | = <yes/no>                    |
| ext-replace         | = <*.extension>               |
| reroute-to          | = <e-mail_address>            |
| smtp-reply-on-block | = <error message>             |

**enable = <yes/no>**

Enable/disable filter.

**action = <forward/block/reroute>**

The action which will be performed on mail/attachment(file) in case of filtering.

forward - forwarding the mail/attachment(file)

block - blocking the mail/attachment(file)

reroute - mails could be rerouted to a specified address. The program rewrites all the RCPT TO fields' value according to the specified rule/value of 'reroute-to' option (it doesn't modify the value of 'To' field). Other filter options will be inactive in case of using this action with the exception of 'header-modify' and 'rewrite' (if you use this action in virus filter template, the VAMS will not kill the known virus just forward the mail).

Example in virus filter: the infected mail will be rerouted to the localhost and the VAMS will insert the virus name into the header subject.

```
action          = reroute
reroute-to      = %rcptuser%@localhost
header-modify   = yes
rewrite         = +X-Virus: %virusname%
```

Find the description of 'reroute-to' option below.

**messages = <label1;label2;...>**

You can specify the labels of the messages which will be sent in case of filtering mail/attachment(file).

**quarantine-copy = <yes/no>**

This field defines whether the program makes a copy of the original of the filtered file. If the 'header-modify' option is enabled, the mail's header will be modified according to the 'rewrite' option's value if 'block' or 'modify' action was taken.

**quarantine-path = <quarantine\_directory\_path>**

You can specify the quarantine directory. Tokens can also be used in this option. Important!

If the value of the quarantine-path option ends in a token that may contain several value (for example: the %recipient% token in case of several recipients) then the mail will be quarantined as many times as the number of the token's possible values. In such a case, the quarantine path consists of a static part (which are the characters found before the token) and one of the token's value from the resolved token list. If the mail is quarantined in more than one location, new IDs will be generated for each instance.

**header-modify = <yes/no>**

Yes: the header of the mail is modified by the value of the rewrite field.



**rewrite** = <field:text>  
**rewrite+** = <field:text>  
**rewrite** = +<field:text>

The value of the specified field in the header part will be modified according to the option's value. The value specified in +<field:value> form means inserting a new field (with its value) into the header. Tokens can be used in the option's value.

**forward-to** = <e-mail\_address>

If you use the forward action, you can set an e-mail address where the mail is forwarded to.

**forward-copy** = <yes/no>

Yes: if 'modify' action is specified in a filter then in case of an incident a copy of the original mail will be sent to the e-mail set in 'forward-to' option.

**ext-replace** = <\*.extension>

If an attachment is deleted and the program is in transparent mode then the file is changed for an other file with same name but other extension. It contains the warning message. You can specify the extension of that file here.

**reroute-to** = <e-mail\_address>

If you set 'reroute' action, the reroute address should be specified in this option. Tokens could also be used in the value:

%rcptdomain% - domain part of the recipient  
%rcptuser% - user part of the recipient

Domains used in this option must be appeared in the 'rules' section as well to successful forward.

**smtp-reply-on-block** = <error message>

Using DirectScan, if the mail is blocked, the specified error message will be passed to the smtp client. Construct the error message in the way of it is described for the 'max-mail-per-connection-smtp-reply' option.

Default: 556 message (%msgid%) content rejected

## The following options, relations must be used in the DEFAULT templates

-----  
#Required options in each filter template:  
action, messages, ext-replace  
#Required option in the field filter:  
contains  
#Required option in the virusfilter, filefilter, filegate:  
filemask  
#If 'action = forward' or 'forward-copy = yes' is set then the 'forward-to' option is required.  
#If 'header-modify = yes' is set then the 'rewrite' option is required.  
#If 'quarantine-copy = yes' is set then 'quarantine-path' option is required.  
#If 'action = reroute' is set then 'reroute-to' option is required.

Filter entry may be broken by a configuration block or a next filter definition.



## Virus filter

You can specify the filter settings for mails containing infected attachment(s).

[template default] virusfilter

```
enable                = <yes/no>

filemask              = <*.extension1;*.extension2;...>
filemask+            = <*.extension1;*.extension2;...>

action                = <forward/block/kill/delete file>

messages             = <label1;label2;...>
iworm-messages       = <label1;label2;...>
wormbuster           = <yes/no>
search-method        = <fast/strict/full>
header-modify        = <yes/no>
rewrite              = <field:text>
rewrite+             = <field:text>
rewrite              = +<field:text>
forward-to           = <e-mail_address>
forward-copy         = <yes/no>
quarantine-copy      = <yes/no>
quarantine-path      = <quarantine_directory_path>
heuristic            = <yes/no>
macro-delete         = <yes/no>
ext-replace          = <*.extension>
block-encrypted-archive = <yes/no>
containers           = <yes/no>
use-regex            = <yes/no>
block-archive-exploit   = <yes/no>
archive-max-decomp-depth = <value>
block-archive-decomp-depth-exceeded = <yes/no>
smtp-reply-on-block  = <error message>
```

**filemask =**

**filemask+ =**

You can specify attachment's filenames to be scanned. In case of filemask = \* all the attachments will be scanned.

You can use the <NULL> parameter to filter out the files (attachments) being without name (e.g.: filemask = \*.exe;<NULL>;\*.com ).

Take care of capital- and little letters! Do not use space characters between parameters!

**action = <forward/block/kill/delete file>**

If a virus is found you have to specify which action must be done.

```
forward      - forwarding mail
block        - blocking mail
kill         - kill virus from infected file(s)
delete file  - deleting attachment(s)
```

**iworm-messages = <label1;label2;...>**

It is possible to set notify messages in case of i-worm incidents.

**wormbuster = <yes/no>**

The i-worm is a 'virus' (false mail) which has a falsified sender, recipient and its content is not relevant for the recipients. The i-worm mails are blocked automatically if wormbuster = yes.



**search-method = <fast/strict/full>**

The method of the virus scanning.

**heuristic = <yes/no>**

Whether the program performs heuristic scan or not.

**macro-delete = <yes/no>**

Yes: all the macros will be deleted.

**block-encrypted-archive = <yes/no>**

Yes: the action specified in the 'action' option will be performed if password protected attachment detected.

No: nothing special action will be performed if password protected attachment detected.

Important!

If encrypted archive files are accepted without checking their content, some of the known e-mail viruses can find entrance into the system encrypted in password protected (non accessible) files. Some variants of the Bagle family and other trojans are spread in this way. Delivering viruses in encrypted archives is not unusual in the world of malwares.

**containers = <yes/no>**

Yes: scanning in container files (archives, compressed files). The VAMS recognizes the compressed, archived files automatically.

**use-regex = <yes/no>**

Yes: parameters specified in the filemask option will be evaluated as regular expressions. Take care of capital- and little letters selecting this feature!

**block-archive-exploit = <yes/no>**

Yes: Checks if the compressed file is an exploit or not according to the set parameters in the [global] section.

Related options:

[global]/archive-max-decomp-size

[global]/archive-max-decomp-ratio

**archive-max-decomp-depth = <value>**

The program will scan the multi-level archives down to the specified depth. If the program finds more depth levels, it will not scan them.

**block-archive-decomp-depth-exceeded = <yes/no>**

If the depth level set in the 'archive-max-decomp-depth' option is exceeded, the mail will be blocked if this option is enabled (yes).



## File filter

[template default] filefilter

enable = <yes/no>

filemask = <\*.extension1;\*.extension2;...>  
filemask+ = <\*.extension1;\*.extension2;...>

action = <forward/block/delete file>

messages = <label1;label2;...>  
header-modify = <yes/no>  
rewrite = <field:text>  
rewrite+ = <field:text>  
rewrite = +<field:text>  
forward-to = <e-mail\_address>  
forward-copy = <yes/no>  
quarantine-copy = <yes/no>  
quarantine-path = <quarantine\_directory\_path>  
ext-replace = <\*.extension>  
use-regex = <yes/no>  
containers = <yes/no>  
smtp-reply-on-block = <error message>

**filemask =**  
**filemask+ =**

You can specify attachment's filenames to be filtered.

You can use the <NULL> parameter to filter out the files (attachments) being without name (e.g.: filemask = \*.exe;<NULL>/\*.com ).

Take care of capital- and little letters! Do not use space characters between parameters!

**action = <forward/block/delete file>**

If a virus is found you have to specify which action must be done.

forward - forwarding mail  
block - blocking mail  
delete file - deleting attachment

**use-regex = <yes/no>**

Yes: parameters specified in the filemask option will be evaluated as regular expressions. Take care of capital- and little letters selecting this feature!

**containers = <yes/no>**

Yes: the program will check the files also in archives if they are conform to the specified masks. If at least one file in the archive is conform to the masks, the whole archive will be handled as defined in the action option.

Important! This option is only utilizable, if the same option is enabled (containers=yes) in the virusfilter template, too!



## File gate

The file gate is working as invert of filefilter. It only let pass through the attachments having the specified extension.

[template default] filegate

```
enable                = <yes/no>

filemask              = <*.extension1;*.extension2;...>
filemask+            = <*.extension1;*.extension2;...>

action                = <forward/block/delete file>

messages              = <label1;label2;...>

header-modify        = <yes/no>
rewrite               = <field:text>
rewrite+             = <field:text>
rewrite              = +<field:text>
forward-to           = <e-mail_address>
forward-copy         = <yes/no>
quarantine-copy      = <yes/no>
quarantine-path      = <quarantine_directory_path>
ext-replace          = <*.extension>
use-regex             = <yes/no>
containers            = <yes/no>
smtp-reply-on-block  = <error message>
```

**filemask =**

**filemask+ =**

Specify the extensions of the attachments which will be accepted.

You can use the <NULL> parameter to filter out the files (attachments) being without name (e.g.: filemask = \*.exe;<NULL>\*.com ).

Please take care of capital and small letters! Do not use space characters between parameters!

**use-regex = <yes/no>**

Yes: parameters specified in the filemask option will be evaluated as regular expressions. Take care of capital- and little letters selecting this feature!

**containers = <yes/no>**

Yes: the program will check the files also in archives if they are conform to the specified masks. If at least one file in the archive is conform to the masks, the whole archive will be handled as defined in the action option.

Important! This option is only utilizable, if the same option is enabled (containers=yes) in the virusfilter template, too!



## Field filter

The field filter provides extended filtering for certain fields of e-mail's header.

In the fieldfilter template the filtering can be set based on different filter levels. Each level must contain a 'field' option, the operations executed in case of action and the 'contains' field. The scanning will be performed on the set header-field of the mail according to the conditions specified in the 'contains' field.

A newer level is led by the <next-level> keyword.

The general settings of field filter - level independent settings - are found before the '#### --- first level' comment, these must not be specified on the next levels.

```
[template default] fieldfilter
```

```
#### --- common
enable          = <yes/no>
quarantine-copy= <yes/no>
quarantine-path= <quarantine_directory_path>
ext-replace     = <*.extension>
#### --- first level
field           = <subject/from/to/cc/date/mailfrom/rcptto>
contains        = <text_to_filter>
contains+       = <text_to_filter>
action          = <forward/block/modify>
messages        = <labell;label2;...>
header-modify   = <yes/no>
rewrite         = <field:text>
rewrite+        = <field:text>
rewrite         = +<field:text>
forward-to      = <e-mail_address>
forward-copy    = <yes/no>
use-regex       = <yes/no>
smtp-reply-on-block = <error message>
#### --- next level
<next-level>
field           = <subject/from/to/cc/date/mailfrom/rcptto>
contains        = <text_to_filter>
contains+       = <text_to_filter>
action          = <forward/block/modify>
messages        = <labell;label2;...>
header-modify   = <yes/no>
rewrite         = <field:text>
rewrite+        = <field:text>
rewrite         = +<field:text>
forward-to      = <e-mail_address>
forward-copy    = <yes/no>
use-regex       = <yes/no>
smtp-reply-on-block = <error message>
```

### **field = <subject/from/to/cc/date/mailfrom/rcptto>**

The field's name on which the searching is performed.

|          |   |
|----------|---|
| subject  | - search in subject field containing the mail's subject |
| from     | - search in 'from' field containing sender's address    |
| to       | - search in 'to' field containing receiver's address    |
| cc       | - search in 'cc' field                                  |
| date     | - search in 'date' field                                |
| mailfrom | - search in 'mailfrom' field                            |
| rcptto   | - search in 'rcptto' field                              |



## Vexira Antivirus for Mail Server

**contains =**

**contains+ =**

Specify the text to be filtered. Please take care of capital and small letters!

**action = <forward/block/modify>**

If a virus is found you have to specify which action must be done.

forward - forwarding mail

block - blocking mail

modify - mail's subject field will be modified if 'header-modify = yes' is set

**use-regex = <yes/no>**

Yes: parameters specified in the contains option will be evaluated as regular expressions. Take care of capital- and little letters selecting this feature!



## Spam filter

The VAMS made for mailing systems to protect them against virus attacks and now it is provided with SPAM filtering as well to protect you and your computer against unsolicited mails.

The spam filter is working based on the spam database(s).

The SPAM filter is activated if the following message can be found in the log file:

```
***** Using Bayes database 'X.X.X.X'
(where 'X.X.X.X' is the version of the spam database)
```

Simple- and multi-level spam filtering can be realized in the spam filter. In case of multi-level spam filtering, you are able to set the filter sensitivity on 3 levels and different actions can be set for each level In case of simple spam filtering, you can set one level and one action.

### ----- Multi-level filtering -----

You can set the spam filter settings in the "[template default] spamfilter" section of the configuration file. If you want to use multi-level spam filtering, you should specify the following settings:

```
[template default] spamfilter

enable                = <yes/no>
quarantine-copy       = <yes/no>
quarantine-path       = <quarantine_directory_path>
whitelist-bypass     = <yes/no>
ext-replace           = <*.extension>
spam-custom-learn    = <yes/no/headermark>
#### --- 1st level
action                = <forward/block/modify>
messages              = <label1;label2;...>
forward-to            = <e-mail_address>
forward-copy          = <yes/no>
header-modify         = <yes/no>
rewrite               = <field:text>
rewrite+              = <field:text>
rewrite               = +<field:text>
smtp-reply-on-block  = <error message>
#### --- 2nd level
<next-level>
action                = <forward/block/modify>
messages              = <label1;label2;...>
forward-to            = <e-mail_address>
forward-copy          = <yes/no>
header-modify         = <yes/no>
rewrite               = <field:text>
rewrite+              = <field:text>
rewrite               = +<field:text>
smtp-reply-on-block  = <error message>
#### --- 3rd level
<next-level>
action                = <forward/block/modify>
messages              = <label1;label2;...>
forward-to            = <e-mail_address>
forward-copy          = <yes/no>
header-modify         = <yes/no>
```



```
rewrite          = <field:text>
rewrite+        = <field:text>
rewrite         = +<field:text>
smtp-reply-on-block = <error message>
```

Options found prior to the '#### --- 1st level' line affect the whole filter globally, so these can't be adjusted on each level.

### **whitelist-bypass = <yes/no>**

Redefine the action to be performed.

Yes: if mail's parameters are equal with one of the White list's entries then the mail will be forwarded without any modification.

### **action = <forward/block/modify>**

Action in case of incident found.

```
forward      - mail is forwarded to a specified recipient set in the forward-to field
block        - blocking mail
modify       - mail's subject field will be modified if 'header-modify = yes' is set
```

### **spam-custom-learn = <yes/no/headermark>**

Custom learning function.

yes: Automatic learning function enabled for the specified template,

mails will be marked by appending the string [VASCL:<scldid>] to the Subject header.

no: Option is inactive (default).

headermark: Learning enabled, mails delivered by the specified template will be marked inserting an extra field (X-VASHLD-SCLID) into their header. See the 'Custom learning' section for more information!

If you place the <next-level> keyword in the template, the multi-level filtering will be activated. A newer (next) level can be led by this keyword.

The general settings of the filter - level independent settings - are found before the '#### --- 1st level' comment, these options must not be specified on the next levels.

In case of multi-level filtering the incoming mails will be filtered on the specified levels. Each level returns its own result (if the mail is spam, or not) then these ones will be evaluated by system. In the course of evaluation the system selects the level on which the mail was marked as spam and performs the action assigned to the level. If the mail is marked on more than one level, the lowest level's action will be performed on the mail. (The 1st level is the lowest one.)

Sensitivity level for the spam levels in case of using the built in spam filter:

```
[template default] spamfilter
#### --- 1st level
low
#### --- 2nd level
<next-level>
normal
#### --- 3rd level
<next-level>
high
```

Explanation of the sensitivity levels can be read in the "Simple filtering" section.

### **Example:**

Setting with 2 levels (only the important options are indicated):

```
[template default] spamfilter
```



```

enable                = yes

#### --- 1st level
action                = block

#### --- 2nd level
<next-level>
action                = modify
header-modify        = yes
rewrite              = SPAM found

```

In this case the 1st and the 2nd level of spam filtering is activated. All two filters' results will be used to evaluate the final result. Those level's settings will be performed which recognize the mail as spam. If more levels recognized the mail as spam, than the lowest level's settings will be applied.

**I.**  
If the mail is not spam according to the 1st level but it is spam according to the 2nd level then the 'modify' action will be performed so the mail's subject will be rewritten by the 'rewrite' field.

**II.**  
If the mail is spam according to both levels, then the 'block' action will be performed because it is in the lowest level.

**Attention!**

If you use multi-level spam filtering you shouldn't use the 'level=' option because it results in simple filtering mode and only the first level's settings will be considered. The filter sensitivity can be controlled by the determined levels in multi-level mode!

-----  
**Simple filtering**  
-----

```

[template default] spamfilter

enable                = <yes/no>
quarantine-copy      = <yes/no>
quarantine-path      = <quarantine_directory_path>
whitelist-bypass     = <yes/no>
ext-replace          = <*.extension>
level                = <low/normal/high>
spam-custom-learn    = <yes/no/headermark>
smtp-reply-on-block  = <error message>

action               = <forward/block/modify>
forward-to           = <e-mail_address>
forward-copy         = <yes/no>
messages             = <label1;label2;...>
header-modify        = <yes/no>
rewrite              = <field:text>
rewrite+             = <field:text>
rewrite              = +<field:text>

```

**level = <low/normal/high>**

Filter sensitivity.

In case of the built in spam filter, the name of the levels indicates a sensitivity level, namely 'high' means the very sensitive level which catches also the spam-suspicious mails. Detailed explanation for the levels:

low



It determines the mail as spam if it is spam with complete certainty. No - or minimal - false positives. It is optimized for really low false positives.

normal

General level, it ensures relatively low level of false positives but the spam recognizing is still on higher level.

high

More improved spam recognizing but the number of false positives will increase.

**action = <forward/block/modify>**

Action in case of spam incident.

forward - mail is forwarded to a specified recipient set in the forward-to field

block - blocking mail

modify - mail's subject field will be modified if 'header-modify = yes' is set

In case of using spam filter in simple mode, the filter sensitivity can be specified on 3 levels (level = low/normal/high) using the 'level' option.

Do NOT use the <next-level> keyword together with 'level' option. The specified sensitivity values means the following levels:

If the mail is marked as spam by the selected level, the specified action will be performed. Different levels should have another actions. Recommended actions for the levels:

- low: block

- normal: forward

- high: modify

### Create additional (custom) spam database

By the help of the tools (mctool, dbtool) you can create additional spam databases which ensure more effective and customized SPAM recognition in co-operation with the official spam database. In the custom spam database you can collect the typical characters and features of your own spams and also using this additional spam database the system will be reach more effective SPAM filtering. Additional spam databases can be specified in the 'additional-sdb' [global] option.

Manuals of the tools can be found in the VAMS package.

### Automatic learning and additional options

The program is able to increase the filter efficiency and upgrades the spam database automatically providing more performance and less false positives.

Not only spam samples but also normal, non-spam ones are needed for the exact spam recognition. VAMS is able to gather and process non-spam mails in the local system and stores them in a separated database (daily.sdb), then this database is used together with the official spam database that ensures more efficiency for spam filtering.

#### Attention!

The spam filter's automatic learning function is designed to be used in enterprise environments so it is not recommended to be used by providers and other major users (e.g. ISPs). Errors may occur due to high loads so Vexira Antivirus supports the automatic learning function's deployment only in enterprise environments and can only provide support if the software has been applied as it is recommended by Vexira Antivirus.

Operation:

First the program collects mails declared as non-spam in the memory and after a specified time-interval it appends them to the existing collection (daily.sdb) and optimizes the database. The operation can controlled in the configuration file by the following options (placed in the [global] section):

**spamal-update-timer = 12h**

The program appends the database from the memory to the daily.sdb after the specified interval. Specifying method: 10h 5m 12s

**spamal-prob-limit = 0.1**

Those mails will get into the database which is declared as spam under the specified rate. Specifying method: percent (0.1 = 10 percent)

**spamal-nospam-total-ratio = 0.6**

The spam and non spam mails ratio will be the specified value. The defined value means the rate of the normal mails. Specifying method: percent (0.1 = 10 percent)

**spamal-ageing-interval = 168**

Additional optimizing of the database will be performed after the specified interval (hours).

Optimizing options of auto learning will be described later.

**Custom learning**

-----

Enable this function in the spam filter template. It is possible to reach higher filter efficiency as follows:

Essentially, spam filter applications can make two mistakes during operation:

1. they do not filter out a real spam, so it is forwarded to the user
2. they mark a non-spam mail as spam (false positive), so the recipient possibly does not get it (this can be serious problem)

VAMS has a solution to reduce the occurrence of the two cases mentioned before:

You can set two 'virtual' e-mail addresses to which users can send the mails recognized incorrectly (as in 1st and 2nd cases). Let the address specified as spam@centralcommand.com for the 1st case and falsepos@centralcommand.com for the 2nd case. These addresses will be handled specially, these are not real e-mail addresses for the VAMS. Mails sent to any of the specified addresses will stay in the system and the VAMS builds a database from them (client.sdb). They will be put into the database as spam or non-spam according to the used e-mail address. This database is also used together with the others providing much more reliability.

Operation:

The program collects these mails in the memory and after a specified time-interval it appends them to the existing database (client.sdb) and optimizes the data.

As the original mail is modified during the receiving processes of the mail server programs, the VAMS stores all the mails in their original form in a special directory (if the function is activated). Thus the VAMS is able to obtain the original mail based on an identifier of the sent mail attached to the 'Subject' field and it will be used for database expansion. The size of this directory is adjustable in the configuration file. The operation can be controlled by the following options (placed in the [global] section):

**spamcl-spam-address = <e-mail\_address>**

Virtual e-mail address for spams, which will get into the database with the specified weight.

**spamcl-nospam-address = <e-mail\_address>**

Virtual e-mail address for non-spams, which will get into the database with the specified weight.

**spamcl-update-timer = 24h**

The VAMS will append the database from the memory to the client.sdb file. Specifying method: 10h 5m 12s

**spamcl-limit-size = 16M**

Size limit of the original mails stored. Specifying method: 16M 48k 900

**spamcl-limit-num = 1024**

Maximum number of the original mails stored (number of pieces).

**spamcl-limit-age = 168h**

Time limit. Mails which are older than the specified time-interval will be deleted. Specifying method: 10h 5m 12s

**spamcl-false-positive-weight = 9**

Those mails which were recognized as spam, but they are normal mails (false positives) will get into the database at the specified weight.

**spamcl-not-detected-weight = 3**

Those mails which were recognized as non-spam, but they are spams will get into the database at the specified weight.

**Weight calculation:**

The spam scan engine analyses the words of new mails one by one. If one of them is found in the spam database, it calculates its SPAM probability (how often this word usually occurs in spam mails), based on the available data stored in the database on that word. This value can be between 0 and 1. The program checks the difference from the neutral value (the neutral words' value is 0.5) then it multiplies the twofold of the difference with the weight value specified by the user. The result will be rounded and the probability of the given word will be increased in the database with this result.

**spamcl-subject-spaces = 0**

The VAMS mail identifier's distance (in characters) from the original subject (the specified distance will be filled with spaces). If this option is not specified, the distance is 0.

**spamcl-remove-id = no**

Removes the custom learn (spamcl) id from the mails marked by the VAMS.

Yes: Deletes the spamcl id from the mail's header or subject field. If spamfilter=enable and spam-custom-learn=(yes|headermark) the new spamcl id will be inserted to the subject field.

No: VAMS id will not be deleted except if spamcl and spam filter functions are enabled (spamfilter=enable and spam-custom-learn=(yes|headermark))

This time the old id will be replaced with a new one.

**Optimizing options settings:****spamal-min-token-count = 1****spamcl-min-token-count = 1**

A token will be removed from the database if it has less instances in the database than the specified value.

**spamal-max-token-size = 30****spamcl-max-token-size = 30**

Maximum size of the tokens (character). During optimization those tokens which consist of more characters than the specified value will be removed from the database.

**spamal-min-token-size = 3****spamcl-min-token-size = 3**

Minimal size of the tokens (character). During optimization those tokens which consist of less characters than the specified value will be removed from the database.



**spamal-max-html-token-size = 50**

**spamcl-max-html-token-size = 50**

Maximum size of the tokens (character) in case of HTML mails. During optimization those tokens which consist of more characters than the specified value will be removed from the database.

**spamal-min-html-token-size = 6**

**spamcl-min-html-token-size = 6**

Minimal size of the tokens (character) in case of HTML mails. During optimization those tokens which consist of less characters than the specified value will be removed from the database.

**spamal-max-prob = 0.7**

**spamcl-max-prob = 0.85**

A token will be considered if its rate is above the specified. This value can be between 0 and 1. Specifying method: percent (0.1 = 10 percent)

**spamal-min-prob = 0.3**

**spamcl-min-prob = 0.15**

A token will be considered if its rate is under the specified. This value can be between 0 and 1. Specifying method: percent (0.1 = 10 percent)

**spamal-max-size = 16M**

**spamcl-max-size = 16M**

Size limit of the database after optimization. Specifying method: 16M 48k 900



## Language filter

The language filter provides great ability to filter e-mails according to the language and script-type of their text parts.

The language/script-type database that needs for the recognition is built in the vexira.sdb (spam database) file so you need to have the spam database file downloaded and available to activate the language filter.

The 'language' means the natural language of the mail-text, for example: English, Hungarian, Russian, Chinese, etc.

The 'script-type' means the character-set used in the mail, for example: Latin, Cyrillic, Greek letters, Far-Eastern letters, etc.

The language filter works based on heuristics recognition so its result will be the most likely script-type/language used in the mail.

If a language/script-type set in the 'mask' option is recognized, the specified action will be performed on the mail.

[template default] languagefilter

```
enable           = <yes/no>
mode             = <inclusive/exclusive>

mask            = <script[/language];script[/language];...>
mask+           = <script[/language];script[/language];...>

action          = <forward/block/modify/reroute>
messages        = <label1;label2;...>
quarantine-path = <quarantine_directory_path>
quarantine-copy = <yes/no>
header-modify   = <yes/no>
rewrite         = <field:text>
rewrite+        = <field:text>
rewrite         = +<field:text>
forward-to      = <e-mail_address>
forward-copy    = <yes/no>
ext-replace     = <*.extension>
whitelist-bypass = <yes/no>
smtp-reply-on-block = <error message>
```

**enable = <yes/no>**

Enable/disable filter.

**mode = <inclusive/exclusive>**

inclusive: only the mails written on the specified language/script-type will be received.

exclusive: the mails written on the specified language/script-type will NOT be received.

Default: inclusive

**mask = [script[/language]];[script];...**

Set the script-type (script) and the language you want the VAMS to scan for. The specified mask-value is considered as regex expression.

The language/script-type database is increasing continuously so you are recommended to use the 'vams languagelist' command to see the currently available languages/script-types.

You can set the script-type and the language separately or together in this options. For example: Latin;English;Latin/English



## Black/White list, RBL

### Black list

-----  
In some cases it is required that the system does not accept mail from a specific address or domain. These can be uninvited advertisements (spam) or uninvited addresses which can be filtered by the help of the following options.

There are two methods to filter incoming mails:

```
# Based on the ip address/address range in the [ip-blacklist] section
If the connected client's ip address is in the given range, the connection will be
terminated by the daemon. The client can't connect from an uninvited ip address.
# Based on the incoming mail's sender in the [mailfrom-blacklist] section
If the mail from: field is equivalent to one of the mailfrom black list's entries
then the connection will be terminated and the sender will get the "invalid request -
domain not valid" error message.
```

It is possible to make rules for outgoing messages as well:

```
# Outgoing (recipient) addresses can be specified to which sending is prohibited in
the [recipient-blacklist] section.
```

Exclusive rules can be defined in the configuration file:

```
[ip-blacklist]
# IP / IPMASK          #
192.168.2.115/255.255.255.255
# or
192.168.2.115/24
```

```
[mailfrom-blacklist]
# domain              #
feri@sulinet.com
sulinet.com
*kovacs@sulinet.com
*sulinet.com
```

```
[recipient-blacklist]
# recipient           #
helen@sulinet.com
domain.net
```

In case of all three black list filter methods a receipt is given by the system containing the fact of filtering in the form of an error or warning message.

In mailfrom-blacklist and recipient-blacklist the \* character can be used at the beginning of the specified name (for example: \*.ceg.com, \*@ceg.com).

### White list

-----  
Operation of this function is to override the blacklist if the client's IP address or the domain or the recipient is equivalent to any of the values specified in the white lists.

So  
# the IP whitelist overrides the ip blacklist and RBL  
# the mailfrom whitelist overrides the mailfrom blacklist  
# the recipient whitelist overrides the recipient blacklist

```
[ip-whitelist]
[mailfrom-whitelist]
```



### [recipient-whitelist]

Specify them in the same way as the exclusive rules.

### RBL (Realtime Blackhole List)

-----  
RBL is a blacklist based on DNS resolving for which you have to specify a domain suffix (you are allowed to specify more than one domain suffices).

Attention!

The default DNS server must be configured to be able to resolve names like:

4.3.2.1.cbl.abuseat.org

(In more detail: <http://cbl.abuseat.org>)

RBL can be defined in the configuration file:

### [ip-rbl]

#domain-suffix #

sbl.spamhaus.org

cbl.abuseat.org



## [Rules] Routing and rules

Routing is a rule system for providing route definition. In this case it defines the method of delivering an smtp (e-mail) by deciding, whether an smtp can enter the given system or whether it should be forwarded to another route. Routing defines the scanning and filtering actions, which should be performed on the given smtp before it enters the system.

If the sender and the recipient of an smtp are known, a rule system can be created, in which the letter's forwarding route and the operations that should be carried out on it can be set.

The routing mechanism can be created using rules. A rule contains a single command, which defines the operations that should be performed on an incoming letter in cases where both the sender and the recipients meet the given criteria. If they do, the rule is activated and actions defined in it will be carried out.

Each rule is written down in a template, in which it can be defined, as to what operations should be performed on the letter and what actions should be taken if it meets one of the filtering criteria. A template is, therefore, a description of filtering rule system.

You can specify the rules in the [rules] section of the configuration file. Each line in this section contains a new rule.

```
[rules]
```

```
Domain/User/Host ; Net/Netmask ; Template ; Target
```

Explanation:

**Domain/User/Host**

-----

On the basis of this value the system applies the rule to the mails in which the mail's receiver has exactly the same domain/user/host as the specified one. You can determine rules for individual e-mail addresses, too.

Using the \* character, the rule will be applied on all incoming mails.

It is possible to use the \* character in the domain/user/host part as well.

For example:

```
sthg.com; 1.2.3.0/255.255.255.0; template; target;
```

```
*sthg.com; 1.2.3.0/255.255.0.0; template; target;
```

If the sthg.com domain is specified, the rule will be applied to addresses ending with <sthg.com> (for example: <user@sthg.com>), but it will be no longer applied to for example <user@other.sthg.com>.

After specifying \*sthg.com the rule will be applied to for example user@other.sthg.com > as well if there are no further rules, as for example:

```
other.sthg.com; 1.2.3.4/1.2.3.4; template; target;
```

or

```
*other.sthg.com; 1.2.3.4/1.2.3.4; template; target;
```

If these rules exist, the they will be preferred.

The precedence of expressions without \* character is higher than one's containing \* character.

Attention! The \* can be specified at the beginning of the domain/user/host.



## Net/Netmask

-----

This value determines a netmask to the specified domain. If sender's IP address matches NET address using netmask then template's rule will be applied to the mail. Otherwise it will be refused. Format of the netmask:

```
<start IP address>/<netmask_length>
```

If there is no rule among entries which matches an incoming mail, the mail will not be processed. If there are more rules determined with same domain/user/host then the system will apply the first one which matches receiver's domain.

## Template

-----

You can specify the label of the templates. Settings under this label will be applied to the mail if the rule is activated. You can specify:

```
# your own template label
```

In this case your own template-settings will be applied according to the specified label.

```
# you don't need to specify template
```

If you don't specify template, then the default one will be applied.

```
# passthrough label
```

In this case there will not be applied any filter. You can make the same function if you switch off all the filters in a template (all the filters contain the enable=no settings).

Default filter settings must be placed in the configuration file. If there are more templates specified with the same name the system will apply all these templates in order.

## Target

-----

You can specify the label of the target. After processing a mail it will be forwarded to the address defined by the target symbol. Target is a symbol similar to Template.

Example:

### Rules1:

```
ide.com ; 192.168.2.0/24 ; virus ; smarthost
```

```
# If a mail comes from 192.168.2.10 IP address and its recipient is somebody@ide.com then this mail will be accepted and processed by virus template and will be forwarded to the mail server symbolized smarthost label
```

```
# If a mail comes from the same IP address but its recipient is somebody@oda.com then this mail will be refused.
```

```
# If a mail comes from 192.171.2.10 IP address and its recipient is somebody@ide.com then this mail will be refused, too.
```

### Rules2:

```
ide.com ; 192.168.1.0/24 ; virus1 ; smarthost
```

```
ide.com ; 192.168.1.0/16 ; virus2 ; smarthost
```

```
# If a mail comes from 192.168.3.4 IP address and its recipient is somebody@ide.com then this mail will be accepted and processed by virus1 template.
```

```
# If a mail comes from 192.168.1.10 IP address and its recipient is somebody@ide.com then this mail will be accepted and processed by virus1 template. (This mail matches both rules but only the first one will be processed.)
```

## Warnings generated by VAMS

-----

The same mechanism determines the targets of the messages generated by the VAMS (warning messages, periodic statistics) with the following additions:

In case of rules which have IP restriction (their NET/NETMASK in not \* ) you should specify a new rule to the domain/host containing 127.0.0.1/32 NET/NETMASK value so that the warning messages can be delivered surely to the sender, too.

Demonstration:



The following rules are available:

```
*.sthg.com;      *;          templat1;   target_sthg
*.vlab.com;     1.2.3.0/24;  template2;  target_vlab
```

For example, a mail is sent from \*.vlab.com (from 10.20.30.1 IP address) to the \*.sthg.com domain. Let the recipient be somebody@sthg.com! A virus is found in the mail so the VAMS has to send a warning message to the recipient and the sender.

If a warning message is sent, its IP address will be equal to the sender's IP address.

The recipient get the warning message without any problem because it is allowed to send mails to the sthg.com domain without IP restriction (its NET/NETMASK is \* ) But it is possible that the mail sent to the sender may not be delivered because there are no rules which match the mail. Because the IP address of the mail is 10.20.30.1 and there is no rule allowing the mail to be delivered from the 10.20.30.1 IP to the vlab.com domain.

In these cases (if the mail is fail to deliver this way) the VAMS gives the 127.0.0.1 IP address to the mail and tries to deliver it again. That's why you need to specify a new rule to the domain/host with 127.0.0.1/32 IP restriction (NET/NETMASK).

The warning mail will be delivered to the sender as well by the following rules:

```
*.sthg.com;      *;          templat1;   target_sthg
*.vlab.com;     1.2.3.0/24;  template2;  target_vlab
*.vlab.com;     127.0.0.1/32;  template2;  target_vlab
```

## Special rule defining method

There is a special rule defining method, in which the same NET/NETMASK, TEMPLATE and TARGET settings can be assigned to different domain addresses. In this case, the domain addresses are read from a list in a file. In the file a line can only contain one domain address. This method of definition is the same as listing the given rule setting with the different domain addresses in the configuration file. Joker characters can't be used in this case so the whole domain and host name must be specify (which is after the @ sign in e-mail addresses)!

The definition is the following:

```
<filename>; NET / NETMASK; TEMPLATE; TARGET
```

In case of a file opening error, the "Cannot open domain-list file:" error message will be returned with the type of the error.



## LDAP support

The LDAP (Lightweight Directory Access Protocol) server is an object oriented database server which is suitable for authentication, too. Mainly it is used for recording of persons which can be arranged in groups. A record can consist of the following data: name, address, phone number, e-mail address, classification, etc. and a personal identifier (password). As it is possible to assign a password to each object, using of LDAP for authentication is an excellent solution. So, it is possible to reduce the risk of illegal using of your mail server. The unauthorized users will not be able to use the system and the authorized ones can be allowed to send e-mails from outside of the network as well.

To activate LDAP features you must install LDAP client libraries from the package 'openldap', which is available for all major Linux distributions and for FreeBSD and Sun Solaris systems as well.

The 'openldap' package can be downloaded from the <http://www.openldap.org> website. You can download and activate the package needed for the VAMS's LDAP support as follows:

- Download the openldap-<version>.tgz file
- Unpack it
- Enter the directory (cd openldap-<version>)
- ./configure --disable-slapd
- make
- make install

After you have done these steps, LDAP support is available for the VAMS.

### Attention!

VAMS LDAP support is available if openLDAP version 2.0.23 or newer is installed! If you use version 2.2.xx of LDAP or libldap.so.2 or liblber.so.2 is missing according to VAMS then make a symlink which points at libldap.so and liblber.so as follows:

```
ln -s libldap.so libldap.so.2
ln -s liblber.so liblber.so.2
```

## LDAP authentication

There is a (limited) support for LDAP-based SMTP authentication in the VAMS. To configure the authentication please add the following options to the 'connector' section of the configuration file:

**ldap-auth-server = your\_server:port**

The format of the value is: host:port

You can specify a name or an IP address as 'host'. If there is no port number specified, the default will be the No. 389 port.

Default: localhost:389

**ldap-auth-prefix = cn**

Default: cn

**ldap-auth-basedn = xxxx**

There is no default value for this setting.

**ldap-auth-timeout = 0**

The time interval while the VAMS waits for the answer of the LDAP server (in seconds). If the value is 0 (zero) then there is no time limit, it waits forever.)

**require-authentication = yes**



Yes: SMTP authentication is forced. If it is set, the VAMS tries to load the 'libldap' and 'liblber' libraries. If they are placed in a directory which is not set for the dynamic linker then you should specify the used directory in the LD\_LIBRARY\_PATH variable before the VAMS is started.

For example:

```
LD_LIBRARY_PATH=/usr/local/lib
export LD_LIBRARY_PATH
```

If the LDAP libraries are not present, then all the connections will be rejected (No. 531 error) unless they are come from a trusted IP (set trusted-ip section).

### [trusted-ip]

[trusted ip] section. Clients from the specified domains don't have to be authenticated. But if they try to authenticate and the login will not be successful, they will be refused.

Example:

```
[trusted-ip]
192.168.2.115/255.255.255.251
# or
127.0.0.1/32
# or
127.0.0.1
```

Currently only PLAIN and LOGIN authentication methods are supported, and the user will be considered authenticated if the LDAP bind is successful with his user name and password (bind-auth).

At this moment we support only one authentication server.

Example for LDAP settings

-----

If you store your users' data as 'cn=username,ou=vams,dc=your,dc=organisation,dc=com' then the VAMS settings shall be:

```
ldap-auth-server      = server.your.organisation.com
ldap-auth-prefix      = cn
ldap-auth-basedn      = ou=vams,dc=your,dc=organisation,dc=com
ldap-auth-timeout     = 0
require-authentication = yes
```

### Get settings from LDAP

LDAP support is available for the following settings:

- Settings in Global section, exceptions are:
  - default-input
  - default-logfile
  - spooldir
  - tempdir
  - connector
  - libdir
  - virus-statistic-log
  - found-viruses-log
  - virus-top-list-log
  - username
  - serialno
  - additional-sdb
- Settings in Connector section, exceptions are:



```

recv-max-proc, recv-load-avg,
recv-load-reg-start
recv-load-reg-end
recv-reg-proc-max
re-sending-X
send-io-error
send-smtp-4xx-error
send-smtp-5xx-error
send-smtp-other-error
send-mailloop
send-internal-error
send-max-retry-exceeded
send-io-error-msg
send-smtp-4xx-error-msg
send-smtp-5xx-error-msg
send-smtp-other-error-msg
send-mailloop-msg
send-internal-error-msg
send-max-retry-exceeded-msg
- Target settings
- Filter settings (virusfilter, filefilter, filegate, fieldfilter, spamfilter)

```

### Attention!

**LDAP settings can not be used in multi level filter templates!**

It means that the values are obtained from LDAP if needed. You should use the following setting form:

**<option> = LDAP(datasource,attribute\_name) | default value**

According to the new setting characteristic, the VAMS tries searching for the option's value on the LDAP server. If it is found then VAMS is using it as the value of the option, if it is not found then the default value will be considered, if any.

The first parameter of the LDAP setting is a query ('datasource') which results in a record which contain values. To obtain that values you have to set the following:

#### **LDAP server name (server):**

This can be specified as 'IP:port'. You can set connection dn and password, too. If the last 2 parameters is not specified, the VAMS will connection to the LDAP server as anonymous, so it can not access the protected areas.

#### **base-dn:**

Query condition.

#### **filter:**

Filter setting.

First step is to define the LDAP server:

```

[ldap_server ldap_1]
Ip          = localhost:389
#bind-dn    = cn=admin,dc=company,dc=com
#bind-pass  = password

```

From now, you can refer to the LDAP server as 'ldap\_1'. Port number specification is optional. Its default value is 389.

In the next step, you can specify the datasource:

```

[datasource ds_smarthost]
server      = ldap_1
base-dn     = ou=vams,dc=company,dc=com
filter     = (mail=%rcptuser%@%rcptdomain%)

```



## Vexira Antivirus for Mail Server

You can refer to the datasource as 'ds\_smarthost'. It queries the data from the 'ldap\_1' LDAP server based on the 'ou=vams,dc=company,dc=com' settings filtered as '(mail=%rcptuser%@%rcptdomain%)'.

After specifying LDAP settings, you can set a 'target' as follows:

```
[target sample]
server = LDAP(ds_smarthost, smarthostIP)|127.0.0.1
port   = LDAP(ds_smarthost, smarthostPort)|25025
```

The 'smarthostIP' is a text-type-, the 'smarthostPort' is an integer-type attribute. You can substitute these attribute names for other ones which have the same types as the originals.

### Attention!

In the configuration file you should specify the LDAP server settings before the line in which you refer to it!

### Attention!

To use other settings you may need to configure the LDAP server. The configuring method is described below.

### Configuring Slapd-type LDAP server

You have to inform the LDAP server on the information's structure to be able to store the VAMS-specific settings.

The vams.schema file includes the structure settings, you can find it in the /usr/share/doc/vams/ directory. This file must be registered into the LDAP server's configuration file named slapd.conf in case of an slapd-type LDAP server. The following line should be inserted into the 'Schema and objectClass definitions' section:

```
include /usr/share/doc/vams/vams.schema
```

After restarting the slapd LDAP server, you will be able to upload your settings.

To set up an entirely new database (VAMS-specific database), you need to initialize the data structure. While initializing, the data structure will be registered into the server. Use the va\_init.ldif file to specify settings necessary for the registration (the file's path is /usr/share/doc/vams/va\_init.ldif). After specifying these data, the following command line should be used to perform registration:

```
ldapadd -v -c -x -D $LDAP_ADMIN_DN -W -f va_init.ldif
```

The initialization should be performed only once!

After initialization the required data must be specified based on the example found in the /usr/share/doc/vams/va\_user.ldif file. It is possible to specify more users' data in this file. Use the following command line to upload the settings:

```
ldapadd -v -c -x -D $LDAP_ADMIN_DN -W -f va_user.ldif
```



## Integration into mail servers

The Vexira Antivirus for Mail Server (henceforward called VAMS) could be built into existing mailing system configurations with very small modification. But, it is essential to have good skill in configuration of the given MTA's. It is recommended to consult its documentation before starting the integration of virus protection.

Conventions used in this document:

**"Port=eSMTP"**

Character strings being between quotes represent entries in the configuration files or commands that have to be entered.

**<something>**

An appropriate parameter have to be substituted in place of characters being between "greater than" and "less than" signs or it shows which key should be used. (for example: <tab>).

## Sendmail

### Mailing through SMTP Relay

-----  
Mailing protection realizing by external SMTP client (for example: Outlook).

The mailer system can be switched over to another port by editing /etc/mail/sendmail.cf configuration file.

- It is possible, the file can be found in another location (for example in /etc directory).

Find the line starting with "O DaemonPortOptions=".

- If there are more active lines containing this entry (there is no hash mark character (#) placed at the beginning of the line) then you should modify that line which has additional entry - "Name=MTA", "Name=NoMTA", "Port=smtp" or "Port=eSMTP" - after the "O DaemonPortOptions=".

If the line that you are ready to modify contains the "Port=<something>" parameter then you can set the port number of VAMS instead of <something>.

- If the "Port=<something>" parameter doesn't occur in the line then insert it right after the "O DaemonPortOptions=" separated by comma from the original parameters.

Examples:

Original entry:

```
O DaemonPortOptions=Port=smtp, Addr=127.0.0.1, Name=MTA
```

Modified entry:

```
O DaemonPortOptions=Port=<Outgoing port number of VAMS>, Addr=127.0.0.1, Name=MTA
```

Original entries:

```
O DaemonPortOptions=Name=MTA
```

```
O DaemonPortOptions=Port=587, Name=MSA, M=E
```

Modified entries:

```
O DaemonPortOptions=Port=<Outgoing port number of VAMS>, Name=MTA
```

```
O DaemonPortOptions=Port=587, Name=MSA, M=E
```

After modification please restart sendmail by the following command:  
**/etc/init.d/sendmail restart**



## Qmail

### Mailing through SMTP Relay

Mailing protection realizing by external SMTP client (for example: Outlook).

1)  
If qmail-smtpd is called by inetd (the "qmail" string is found in /etc/inetd.conf file in a valid line starting with the "smtp" keyword):

Add the following line in the /etc/services file:  
**qsmtp <something>/tcp mail**  
(you should replace <something> with the outgoing port number of the VAMS).

Modify /etc/inetd.conf file. In the line starting with smtp you should replace smtp string with qsmtp.

For example the file contains the following line:  
**smtp stream tcp nowait qmaild /var/qmail/bin/tcp-env tcp-env /var/qmail/bin/qmail-smtpd**  
then replace with  
**qsmtp stream tcp nowait qmaild /var/qmail/bin/tcp-env tcp-env /var/qmail/bin/qmail-smtpd**

Restart inetd by the following command:  
**/etc/init.d/inetd restart**

2)  
If qmail-smtp is called by tcpserver (if /etc/inetd.conf file contains valid line starting with smtp string):

in the qmail-smtpd starter script (/etc/qmail/qmail-smtpd/run) you have to set the tcpserver's "smtp" parameter to the outgoing port number of the VAMS.

For example the file contains the following line:  
**tcpserver -v -R -l 0 -x /etc/qmail/tcp.smtp.cdb -c "\$MAXSMTPD" -u "\$QMAILDUID" -g "\$NOFILESGID" 0 smtp qmail-smtpd 2>&1**  
then replace with  
**tcpserver -v -R -l 0 -x /etc/qmail/tcp.smtp.cdb -c "\$MAXSMTPD" -u "\$QMAILDUID" -g "\$NOFILESGID" 0 <something> qmail-smtpd 2>&1**  
(you should replace <something> with the outgoing port number of the VAMS).

Restart qmail by the following command:  
**/etc/init.d/qmail restart**

## Postfix

### Mailing through SMTP Relay

Mailing protection realizing by external SMTP client (for example: Outlook).

You can reconfigure mailserver's port by editing a table found in **/etc/postfix/master.cf**.

Find the following line:  
**smtp inet n - - - smtpd**  
and replace with  
**<something> inet n - - - smtpd**  
(you should replace <something> with the outgoing port number of the VAMS).



Restart postfix by the following command:  
`/etc/init.d/postfix restart`

Using in sandwich mechanism  
-----

This method makes VAMS suitable for protection of the mailer system running on the server (for example: Pine, Mutt) and utilize special functions of it not supported in VAMS yet (for example: e-mail authentication).

### Attention!

**You** should set the `port-banner=no` in the `vams.conf` file, otherwise the postfix returns "mail loop" and the sandwich mechanism will not work.

Insert the following line in the `/etc/postfix/main.cf` file as a new line:  
`content_filter = smtp:[127.0.0.1]:<something>`  
(**you** should replace `<something>` with incoming port number of the VAMS).

Add the following lines in the `/etc/postfix/master.cf` file:

```
localhost:<something> inet n - n - 64 smtpd
-o content_filter=
-o local_recipient_maps=
-o myhostname=VAMS.dummy
-o mynetworks=127.0.0.0/8
-o strict_rfc821_envelopes=yes
```

(**you** should replace `<something>` with the outgoing port number of the VAMS).

The VAMS's configuration file must contain the `'port-banner = no'` setting.

After having done these setting and postfix still returns error (Relay access denied) then you have to set it to accept localhost as a relay.

In the `/etc/postfix/main.cf` file you should modify the line starting with "mynetworks" string. It has to contain the "127.0.0.0/8" string.

- If there is no valid line starting with "mynetworks" then add the following line:  
`mynetworks = 127.0.0.0/8`
- **If there** is line starting with "mynetworks" but it has different value from "127.0.0.0/8" then add it separated by comma.

For example the original was  
`mynetworks = 192.168.2.0/24`  
the you should replace with  
`mynetworks = 192.168.2.0/24, 127.0.0.0/8`

Restart postfix by the following command:  
`/etc/init.d/postfix restart`

## Exim

### Mailing through SMTP Relay

-----

Mailing protection realizing by external SMTP client (for example: Outlook).

- 1)  
If exim is called by inetd  
(the "exim" string is found in `/etc/inetd.conf` file in a valid line starting with the "smtp" keyword):



## Vexira Antivirus for Mail Server

Add the following line in the `/etc/services` file:

```
exsmtp      <something>/tcp      mail
```

(you should replace `<something>` with the outgoing port number of the VAMS).

Modify `/etc/inetd.conf` file. In the line starting with "smtp" you should replace "smtp" string with "qsmtp".

For example the file contains the following line

```
smtp      stream  tcp  nowait  mail  /usr/sbin/exim exim -bs
```

then replace with

```
exsmtp   stream  tcp  nowait  mail  /usr/sbin/exim exim -bs
```

Restart `inetd` by the following command:

```
/etc/init.d/inetd restart
```

2)

If `exim` runs as a daemon

(the `/etc/inetd.conf` file contains valid line starting "with" smtp string):

You can reconfigure mailserver's port by editing the Exim's configuration file (`/etc/exim/exim.conf`).

The following must be entered in the first line of the file (or in the "Main configuration settings" part):

```
daemon_smtp_port = <something>
```

(you should replace `<something>` with the outgoing port number of the VAMS).

Restart `exim` by the following command:

```
/etc/init.d/exim restart
```



## END USER AGREEMENT

IF YOU DO NOT AGREE TO THESE TERMS AND CONDITIONS DO NOT INSTALL OR USE THE VEXIRA ANTIVIRUS SOFTWARE (referred to hereafter as the "Software"). BY CLICKING "YES", "I ACCEPT", "I AGREE", "OK", "CONTINUE", "NEXT" OR BY INSTALLING OR USING THE SOFTWARE IN ANY WAY, YOU ARE INDICATING YOUR COMPLETE UNDERSTANDING AND ACCEPTANCE OF THE TERMS AND CONDITIONS OF THIS END USER SOFTWARE LICENSE AGREEMENT (referred to hereafter as the "License"). IF YOU DO NOT ACCEPT ALL OF THE TERMS AND CONDITIONS OF THIS LICENSE, THEN CENTRAL COMMAND, INC. IS UNWILLING TO LICENSE THE SOFTWARE TO YOU. YOU MAY, WITHIN THIRTY (30) DAYS OF YOUR INITIAL PURCHASE OF A COPY OF THE SOFTWARE, RETURN THE ENTIRE COPY OF THE SOFTWARE (INCLUDING ALL COMPUTER MEDIA, PACKAGING AND DOCUMENTATION) WITH PROOF OF PURCHASE EITHER TO CENTRAL COMMAND, INC. DIRECTLY AT ITS CUSTOMER SERVICE DEPARTMENT OR TO THE RETAILER FROM WHICH YOU PURCHASED THE SOFTWARE, FOR A FULL REFUND OF THE AMOUNT INDICATED BY YOUR SALES RECEIPT OR PROOF OF PURCHASE FOR THE SOFTWARE.

IF YOU ARE INSTALLING THE SOFTWARE ON A COMPUTER THAT IS NOT OWNED BY YOU, YOU ARE BOUND TO THE TERMS AND CONDITIONS OF THIS LICENSE BOTH IN YOUR INDIVIDUAL CAPACITY AND AS AN AGENT OF THE OWNER OF THE COMPUTER, AND YOUR ACTIONS WILL BIND THE OWNER OF THE COMPUTER. YOU REPRESENT AND WARRANT TO CENTRAL COMMAND, INC. THAT YOU HAVE BOTH THE CAPACITY AND AUTHORITY TO ENTER INTO THIS LICENSE ON YOUR OWN BEHALF AS WELL AS ON BEHALF OF THE OWNER OF THE COMPUTER ON WHICH YOU ARE INSTALLING THE SOFTWARE. FOR PURPOSES OF THIS LICENSE, THE "OWNER" OF A COMPUTER IS THE INDIVIDUAL OR ENTITY THAT HAS LEGAL TITLE TO THE COMPUTER OR THAT HAS THE POSSESSORY INTEREST IN THE COMPUTER IF IT IS LEASED OR LOANED BY THE ACTUAL TITLE OWNER.

This End User License Agreement ("License") is a legal agreement between you (either an individual, agent of the owner, or a single entity end user) and Central Command, Inc. for use of the Central Command, Inc. software product identified above (i.e. Vexira Antivirus), which includes computer software and may include associated media, printed materials, and "online" or electronic documentation (collectively referred to as the "Software"), all of which are protected by U. S. copyright laws and international treaty protection. By installing, copying, or otherwise using the Software, you agree to be bound by the terms of this License. If you do not agree to the terms of this License, do not install or use the Software.

The Software and the name "Vexira Antivirus" is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Software is licensed, not sold. If you agree to be bound by all of the terms of this License, you will only own the media on which the Software has been provided and not the Software itself.

**THIRTY DAY MONEY BACK GUARANTEE:** If you are the original licensee of this copy of the Software and are dissatisfied for any reason with it within the first thirty (30) days after your purchase or delivery date, you may return the complete product, together with your original proof of purchase to Central Command, Inc. or the retailer from which you purchased the Software, for a refund of the amount indicated by your original proof of purchase. If this purchase was completed using electronic delivery you are required to complete a Letter of Destruction (referred to hereafter as an "LOD") and return it within thirty (30) days after your purchase date to receive a refund. Central Command, Inc. uses the postmark or fax date of the completed and returned LOD to determine compliance. You can receive a LOD by contacting your electronic retailer from which you purchased the software or directly from Central Command, Inc. via e-mail at [service@centralcommand.com](mailto:service@centralcommand.com), postal mail at P.O. Box 468, Medina, Ohio, 44256, or fax at +1 330-722-6517. For assistance you may also contact Central Command, Inc. by calling +1 330-723-2062 and requesting Customer Service.

**GRANT OF LICENSE:** Central Command, Inc. hereby grants you and only you a non-exclusive license to use the Software subject to and upon all of the terms and conditions set forth in this License.

**APPLICATION SOFTWARE:** You may install and use only one copy of the Software, and only on a single computer terminal.

**NETWORK USE:** You may also store or install a copy of the Software on a storage device, such as a network server, which is used only to install or run the Software on your other computers over an internal network; however, you must purchase and dedicate a separate license for each separate computer terminal on which the Software is installed or run from the storage device. A license for the Software may not be shared or used concurrently on different computers or computer terminals. You are required to purchase a license pack or multi-use license if you require multiple licenses for use on multiple computers or computer terminals.

If you purchase a License Pack and you have acquired this License for multiple licenses of the Software, you may make the number of additional copies of the computer software portion of the Software specified above as "Licensed copies." You are also entitled to make a corresponding number of secondary copies for use on a single home computer as specified above in the section entitled "Application Software".

If you purchase a License for the Software to be used to virus scan electronic messages or you install the Software in such a way to virus scan electronic messages you are required to purchase a license for each domain name and each sub domain name that is virus scanned. If your total electronic mail addresses exceed 6000 you are required to purchase a special Internet



## Vexira Antivirus for Mail Server

Service Provider (ISP) License for use of the Software.

**TERM OF LICENSE:** The License granted hereunder shall commence on the date that you install, copy or otherwise first use the Software. You may terminate this License at any time. This License shall terminate automatically (and you shall have no right to use the Software) upon your breach of any term of this License. Upon termination, you must destroy the Software and all copies, if any, you made pursuant to this License.

**UPGRADES:** If the Software is labeled as an upgrade, you must be properly licensed to use a product identified by Central Command, Inc. as being eligible for the upgrade in order to use the Software. A copy of the Software labeled as an upgrade replaces and/or supplements the product that formed the basis for your eligibility for the upgrade. You may use the resulting upgraded product only in accordance with the terms of this License. If the Software is an upgrade of a component of a package of software programs that you licensed as a single product, the Software may be used and transferred only as part of that single product package and may not be separated for use on more than one computer.

**COPYRIGHT:** All right, title and interest in and to the Software and the name "Vexira Antivirus" and "Vexira" and all copyright rights in and to the Software (including but not limited to any images, photographs, animations, video, audio, music, text, and "applets" incorporated into the Software), the accompanying printed materials, and any copies of the Software are owned by Central Command, Inc. and/or its suppliers. The Software is protected by copyright laws and international treaty provisions. Therefore, you must treat the Software and the term "Vexira Antivirus" or "Vexira" like any other copyrighted material except that you may install the Software on a single computer provided you keep the original solely for backup or archival purposes. You may not copy the printed materials accompanying the Software. You may not use the name "Vexira Antivirus" or "Vexira" or any similar name except when referring to the Software. You must produce and include all copyright notices in their original form for all copies created irrespective of the media or form in which the Software exists. You may not sub-license, rent, sell, or lease the Software. You may not reverse engineer, recompile, disassemble, create derivative works, modify, translate, or make any attempt to discover the source code for the Software or any part thereof. Except as expressly permitted by applicable law, you may not remove from the Software, or alter, any of the trademarks, trade names, logos, patent or copyright notices or other proprietary rights notices or markings, or add any other notices or markings to the Software.

**LIMITED WARRANTY:** Central Command, Inc. warrants that the media on which the Software is distributed is free from defects for a period of thirty (30) days from your date of receipt or purchase date of the Software. Your sole remedy for a breach of this warranty will be that Central Command, Inc. at its option, may replace the defective media upon receipt of the damaged media, or refund the money you paid for the Software. Central Command, Inc. does not warrant that the Software will be uninterrupted or error free or that the errors will be corrected. Central Command, Inc. does not warrant that the Software will meet your requirements. CENTRAL COMMAND, INC. HEREBY DISCLAIMS ALL OTHER WARRANTIES FOR THE SOFTWARE, WHETHER EXPRESSED OR IMPLIED. THE ABOVE WARRANTY IS EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, WHETHER EXPRESSED OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHER RIGHTS, WHICH VARY FROM STATE TO STATE.

**DISCLAIMER OF DAMAGES:** Anyone installing, using, testing, or evaluating the Software bears all risk to the quality and performance of the Software. In no event shall Central Command, Inc. be liable for any damages of any kind, including, without limitation, direct, indirect, exemplary, special, consequential or incidental damages of any kind (including without limitation lost profits or damage to other systems) arising out of the use, performance, or delivery of the Software, even if Central Command, Inc. has been advised of the existence or possibility of such damages. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU. IN NO CASE SHALL CENTRAL COMMAND, INC.'S LIABILITY EXCEED THE PURCHASE PRICE PAID BY YOU FOR THE SOFTWARE. The disclaimers and limitations set forth above will apply regardless of whether you accept or use, evaluate, or test the Software.

**IMPORTANT NOTICE TO USERS:** THE SOFTWARE IS NOT FAULT-TOLERANT AND IS NOT DESIGNED OR INTENDED FOR USE IN ANY HAZARDOUS ENVIRONMENT REQUIRING FAIL-SAFE PERFORMANCE OR OPERATION. THIS SOFTWARE IS NOT FOR USE IN THE OPERATION OF AIRCRAFT NAVIGATION, NUCLEAR FACILITIES, OR COMMUNICATION SYSTEMS, WEAPONS SYSTEMS, DIRECT OR INDIRECT LIFE-SUPPORT SYSTEMS, AIR TRAFFIC CONTROL, OR ANY APPLICATION OR INSTALLATION WHERE FAILURE COULD RESULT IN DEATH, SEVERE PHYSICAL INJURY OR PROPERTY DAMAGE.

**GOVERNMENT RESTRICTED RIGHTS/RESTRICTED RIGHTS LEGEND:** Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of Commercial Computer Software-Restricted Rights clause at 48 CFR 52.227-19, as applicable. Contact Central Command, Inc., at P.O. Box 468, Medina Ohio 44258-0468.

**GENERAL:** This License is deemed delivered in, and will be governed by, the laws of the State of Ohio, in the United States of America. This License may only be modified by a license addendum, which must accompany this License or by a written document which has been signed by both you and Central Command, Inc. This License has been written in the English language only and is not to be translated or interpreted in any other language. Prices, costs and fees for use of the Software are subject to change without notice to you. In the event of invalidity of any provision of this License, the invalidity shall not affect the validity of the remaining portions of this License. Vexira, Vexira logo, Central Command, Central Command's logo, EVRT, Emergency Virus Response Team, Without us, there's no defense, are trademarks of Central Command, Inc. Microsoft,



## Vexira Antivirus for Mail Server

Windows, Excel, Word, the Windows logo, Windows NT, Windows 2000 are registered trademarks of Microsoft Corporation. All other trademarks or tradenames are the property of their respective owners.

### CONTACT

This manual provides comprehensive information on operational of our virus protection product. If you have any additional questions about it or would like to share your experience or proposals with us do not hesitate to contact us! Turn to us with confidence, your demands and remarks will be respected.

Address Central Command, Inc.  
Medina, Ohio 44258,  
P. O. Box 468.  
United States

Phone (+1) 330 723 2062  
Fax (+1) 330 722 6517  
Web <http://www.centralcommand.com>  
Support <http://www.centralcommand.com>  
E-mail [sales@centralcommand.com](mailto:sales@centralcommand.com)  
[support@centralcommand.com](mailto:support@centralcommand.com)