

KASPERSKY LAB

Kaspersky Anti-Virus 5.5 for Proxy Server

ADMINISTRATOR'S GUIDE

KASPERSKY ANTI-VIRUS 5.5 FOR PROXY SERVER

Administrator's Guide

© Kaspersky Lab
<http://www.kaspersky.com>

Revision date: August 2008

Contents

CHAPTER 1. KASPERSKY ANTI-VIRUS FOR PROXY SERVER	6
1.1. What's new in version 5.5?	6
1.2. Hardware and software requirements	7
1.3. Licensing policy	8
1.4. Distribution kit	9
1.4.1. License agreement	9
1.5. Help desk for registered users	10
1.6. Conventions	10
CHAPTER 2. HOW IT WORKS, AND TYPICAL INSTALLATIONS	12
2.1. How the application works	12
2.2. Typical deployment scenarios	15
2.2.1. Installation on the same server with SQUID proxy	15
2.2.2. Installation on a dedicated server	16
CHAPTER 3. INSTALLING THE APPLICATION	18
3.1. Installation on a server running Linux	18
3.2. Installation on a server running FreeBSD	19
3.3. Installation procedure	19
3.4. Post-installation configuration	20
3.5. Distribution of the application files in directories	21
CHAPTER 4. USING KASPERSKY ANTI-VIRUS	24
4.1. Updating the anti-virus databases	24
4.1.1. Automatic updating of the anti-virus databases	25
4.1.2. Manual updating of the anti-virus databases	26
4.1.3. Creating a shared directory for storing and sharing database updates	27
4.2. Managing license keys	28
4.2.1. Viewing information about license keys	29
4.2.2. Renewing your license	31
4.2.3. Removing a license key	32
4.3. Using a control script	32
4.4. Ensuring anti-virus protection of HTTP traffic	33

4.5. Configuring anti-virus scan parameters for user groups	35
CHAPTER 5. DETAILED SETTINGS FOR KASPERSKY ANTI-VIRUS	38
5.1. Creating groups	38
5.2. Anti-virus scan settings.....	40
5.3. Choosing actions for scanned objects.....	40
5.4. Administrator notifications	42
5.5. Operation modes	44
5.6. Modes of interaction with proxy via ICAP	45
5.7. Logging application statistics.....	46
5.8. Application reporting parameters	48
5.9. Creating a memory dump to detect errors.....	50
5.10. Work with Internet broadcasting stations.....	51
5.11. Optimizing Kaspersky Anti-Virus	51
5.11.1. Reducing traffic	51
5.11.2. Setting up exclusions.....	52
CHAPTER 6. UNINSTALLING THE APPLICATION	54
APPENDIX A. APPLICATION REFERENCE.....	55
A.1. <i>kav4proxy.conf</i> application configuration file.....	55
A.2. Macros	62
A.3. <i>kavicapserver</i> return codes.....	63
A.4. Command line options for <i>licensemanager</i>	63
A.5. <i>Licensemanager</i> return codes.....	64
A.6. <i>Keepup2date</i> command line options.....	64
A.7. <i>Keepup2date</i> return codes	66
APPENDIX B. KASPERSKY LAB.....	67
B.1. Other Kaspersky Lab Products	68
B.2. Contact Us.....	78

CHAPTER 1. KASPERSKY ANTI-VIRUS FOR PROXY SERVER

Kaspersky Anti-Virus 5.5 for Proxy Server (hereinafter also referred to as *Kaspersky Anti-Virus* or the *application*) provides anti-virus protection for network traffic routed through proxy servers which support the Internet Content Adaptation Protocol (ICAP) in accordance with RFC 3507.

The application allows the user to:

- Perform anti-virus scans on objects transferred through the proxy server.
- Cure infected objects, or block access to infected objects if disinfection fails.
- Use group settings to define filtration parameters that are applied depending on the address of the user requesting an object, and the object's address (URL).
- Log activity statistics, including information about anti-virus scanning and its results, and application errors and warnings.
- Notify administrators about detection of malicious software.
- Update the anti-virus databases. By default the application uses Kaspersky Lab's update servers as the source of updates, but it can be configured to update the databases from a local directory.

The anti-virus databases are used in the detection and disinfection of infected objects. The application uses database records to analyze every object, checking it for virus presence: its content is compared with code typical for specific viruses.



Please be aware that new viruses appear every day, and therefore you are advised to maintain the anti-virus databases in an up-to-date state. New updates are available hourly on Kaspersky Lab's update servers.

1.1. What's new in version 5.5?

The current version of Kaspersky Anti-Virus has the following improvements:

- Support for 64-bit operating systems added.
- Support for Squid 2.6 has been added

- Support for Cisco Content Engine, BlueCoat ProxySSG and Netcache have been added. For information on deploying the application with this hardware, see 2.2.2 on p. 16.
- New options are available for setting up groups: the maximum scan time and the anti-virus databases set. For more information on setting up groups, see 5.2 on p. 40.
- Support for the ICAP preview feature has been added, which reduces traffic and filtration time. For more information on using the preview feature, see 5.11.2 on p. 52.
- Option of viewing detailed information on the license by traffic is added (see).
- Application performance is increased.

1.2. Hardware and software requirements

To ensure normal functioning of Kaspersky Anti-Virus, the system must meet the following hardware and software requirements:

Minimum hardware requirements for product operation:

- Intel Pentium® 133 MHz processor or higher
- 64 MB RAM
- 50 MB of disk space for application setup
- 200 MB of available disk space for temporary files.

The configuration is intended to service at least 10 clients sending at least 20 requests per minute, with an average request size of 15 Kb.

Optimal hardware requirements:

- for a proxy server servicing requests from 50 clients, with an average load of 900 requests per minute and daily traffic of 250 MB:
 - Intel Pentium® II 300 MHz processor.
 - 128 MB RAM.
 - 512 MB of available disk space for temporary files.
- for a proxy server servicing requests from 250 clients, with an average load of 1300 requests per minute and daily traffic of 1 GB:

- Intel Pentium® 4 processor.
- 512 MB RAM.
- 1 GB of available disk space for temporary files.

Software requirements:

- One of the following operating systems for 32-bit platforms:
 - RedHat Enterprise Linux Advanced Server 4 UPD4
 - Fedora Core 6
 - SUSE Linux Enterprise Server 10
 - openSUSE Linux 10.2
 - Debian GNU/Linux 3.1 updated (r4)
 - Mandriva 2007
- One of the following operating systems for 64-bit platforms:
 - RedHat Enterprise Linux Advanced Server 4 UPD4
 - Fedora Core 6
 - SUSE Linux Enterprise Server 10
 - openSUSE Linux 10.2.
- Squid 2.5, 2.6 and 3.0 proxy server with ICAP support.
- Perl 5.0 or higher (www.perl.org).
- Glibc 2.2.x or higher (for Linux distributions).

1.3. Licensing policy

The licensing policy for Kaspersky Anti-Virus limits product use based on the following criteria:

- number of users protected by the application;
- amount of incoming traffic processed.

The first criterion limits the number of users whose computers' traffic is checked.

The second criterion limits the number of incoming traffic per last 24 hours. Only the traffic, processed by the application is taken into account.

If the license criterion has reached 90% of the limit, the appropriate message is added to the report file. The same notification takes place when the license limit is exceeded.

After the license has expired, the application functions normally but the anti-virus bases cannot be updated.

1.4. Distribution kit

You can purchase the product either from our dealers (retail box) or at one of our online stores (for example, www.kaspersky.com – follow the **E-store** link).

The retail box contains:

- a sealed envelope containing the installation CD with the product.
- a copy of this Administrator's Guide.
- a license key file, either bundled with the distribution package or recorded on a special floppy disk.
- the License Agreement.

If you purchase our application online, you will download it from Kaspersky Lab's website. Your license key is either included in the installation package or will be sent to you by email after payment.

1.4.1. License agreement

The license agreement constitutes a legal agreement between you and Kaspersky Lab Ltd containing the terms and conditions subject to which you may use the purchased software.



Please read the license agreement carefully!

If you do not agree with the terms of the license agreement you may return the box with Kaspersky Anti-Virus to the distributor from which you purchased it. You will be refunded the amount paid for subscription, provided the CD envelope remains sealed.

Opening the sealed envelope of the installation CD or installing the product on a computer constitutes your acceptance of all terms and conditions of the license agreement.

1.5. Help desk for registered users

Kaspersky Lab offers an extensive service package enabling registered customers to boost the productivity of Kaspersky Anti-Virus.

During the time that your license is valid, you will be provided with these services:

- new versions of this software product, provided free of charge
- phone or email support on matters related to the installation, configuration, and operation of the product you have purchased
- notifications about new software products from Kaspersky Lab, and about new virus outbreaks. This service is provided to users who have subscribed to the Kaspersky Lab email newsletter service.








Kaspersky Lab does not give advice on the performance and use of your operating system or other technologies.

1.6. Conventions

Various formatting conventions are used throughout the text of this document depending on the purpose of a particular element. Table 1 below lists the formatting conventions used.

Table 1. Conventions

Style	Meaning
Bold type	Menu titles, menu items, window titles, parts of dialog boxes, etc.
 Note.	Additional information, notes.
 Attention!	Information requiring special attention.

Style	Meaning
 <i>To perform the action,</i> 1. Step 1. 2. ...	Procedure description for user's steps and possible actions.
 Task, example	Statement of a problem, example for using the software features.
 Solution	Solution to a defined problem.
[key] – key purpose.	Command line parameters.
Text of information messages and the command line	Text of configuration files, information messages and the command line.

CHAPTER 2. HOW IT WORKS, AND TYPICAL INSTALLATIONS

This chapter explains the application's functionality, its configuration and integration with an existing network structure.

2.1. How the application works

Kaspersky Anti-Virus performs anti-virus scanning of HTTP traffic using two modes of proxy operation: **REQMOD** and **RESPMOD**.

In the **RESPMOD** mode, the application checks objects requested by users via a proxy server. In the **REQMOD** mode it scans objects transmitted by users through the proxy: for instance, for a web-based mail server interface, Kaspersky Anti-Virus scans message attachments transferred by users to mail servers.

In the **RESPMOD** mode, the application uses this algorithm to scan internet traffic (see Fig. 1):

1. The user requests an object through a proxy via HTTP.
2. If the requested object is available within the proxy cache, it will be returned to the user. If the object is not found in the cache, the proxy accesses a remote server and downloads the requested object from it.
3. The proxy uses ICAP to transfer the retrieved object to Kaspersky Anti-Virus for an anti-virus check.
4. Kaspersky Anti-Virus looks for a correspondence between the request parameters (user IP address, URL of the requested object) and its groups (please refer to section 5.1 on p. 38 for details about groups). If it finds a correspondence, it scans and processes the object in accordance with the rules specified for that group. If a request does not match any of the existing groups, the application uses the default group rules for anti-virus scanning and processing.
5. The application assigns a specific status to a scanned object on the basis of the anti-virus scan results. The status determines whether attempts by users to access the object will be granted or blocked (please refer to section 5.3 on p. 40 for details about available statuses and actions performed by the application). Access to objects with a specific status is

granted or blocked according to the processing group parameters (please refer to section 5.1 on p. 38 for details about groups).

6. If access to an object has been granted, Kaspersky Anti-Virus allows the proxy to cache the object and transmit it to users. If access to an object is blocked, Kaspersky Anti-Virus prevents the proxy from caching the object or delivering it to users. Instead of receiving the requested object, the user will be notified that access to the object has been blocked.

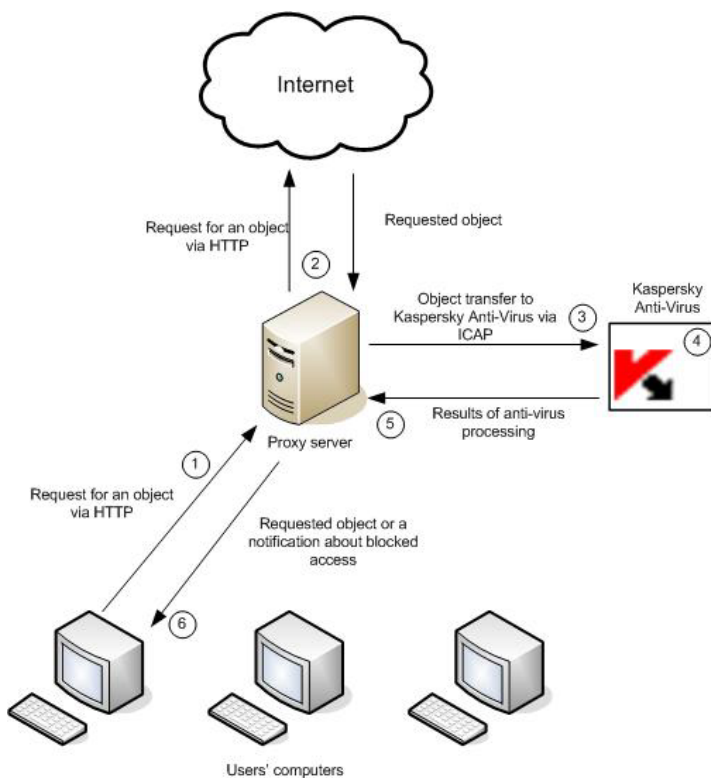


Figure 1. Anti-virus scanning of traffic in the **RESPMOD** mode

In the **REQMOD** mode, the application uses this algorithm to scan internet traffic (see Fig. 2):

1. The user sends an object using HTTP via a proxy.
2. The proxy uses ICAP to transfer the received object to Kaspersky Anti-Virus for an anti-virus scan.

3. Kaspersky Anti-Virus looks for a correspondence between the request parameters and any of the existing groups (please refer to section 5.1 on p. 38 for details about groups). If it finds a correspondence, it scans and processes the object in accordance with the rules for that group. If a request does not match any existing group, the application uses the default group rules for anti-virus scanning and processing.

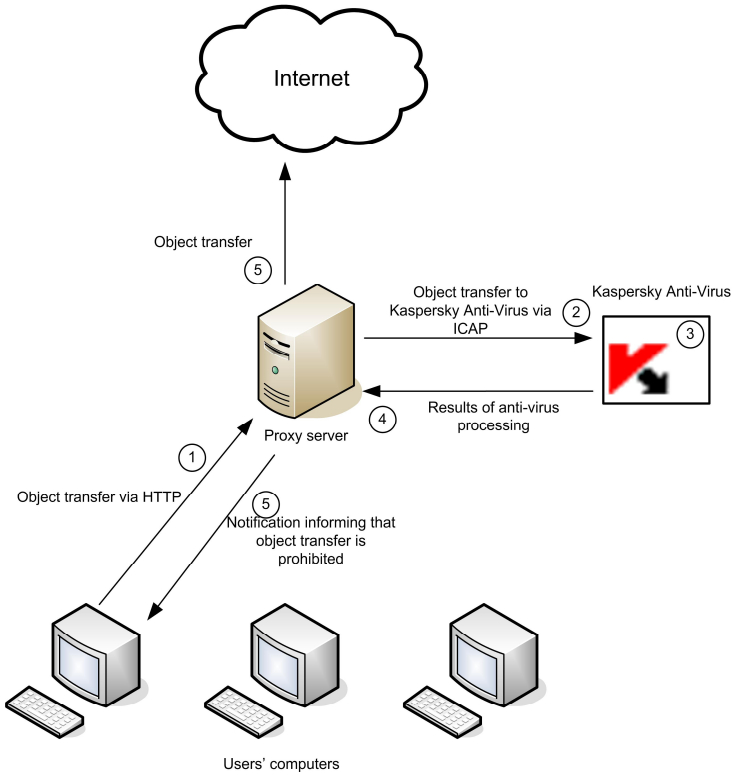


Figure 2. Anti-virus scanning of traffic in the **REQMOD** mode

4. The application assigns a specific status to a scanned object on the basis of the anti-virus scan results. The status determines whether transfer of the object is allowed or prohibited (please refer to section 5.3 on p. 40 for details about available statuses, and actions performed by the application). Permission or denial of transfer for objects with a specific status is defined according to the processing group parameters (please refer to section 5.1 on p. 38 for details about groups).

5. If transfer is allowed, the proxy transmits the object sent by the user. If transfer is prohibited, the proxy does not transmit the object and instead notifies the user that the transfer has been blocked.

2.2. Typical deployment scenarios

This section describes the two main scenarios for deploying the application:

- Application setup on the same server with a proxy.
- Application setup on a dedicated server.

The examples below contain general guidelines, allowing you to configure the application in accordance with your existing network structure.

2.2.1. Installation on the same server with SQUID proxy



Further in this document we shall refer to this variant of Kaspersky Anti-Virus setup (on the same server with proxy) as default for describing the application operation and configuration.

Installing the application on the same server with a proxy allows faster processing, as data transfers between proxy and Kaspersky Anti-Virus occur locally and do not involve the network. This deployment scheme is efficient in the case of a low load on the proxy server. If the proxy services a large number of user requests you are advised to install the application on a dedicated server, since anti-virus scanning and processing are resource-intensive procedures and could negatively influence the proxy's performance. Please refer to section 2.2.2 on p. 16 for installing the application on a dedicated server.

During application setup the installer automatically configures the following aspects:

1. Kaspersky Anti-Virus will be set up to run automatically when the operating system boots, and will listen for requests from the Squid proxy using port 1344 for all the server's network interfaces.
2. The following lines will be added to the **ICAP OPTIONS** section in the Squid configuration file specified during application setup:

```
icap_enable on
icap_send_client_ip on
icap_service is_kav_resp respmod_precache 0
icap://localhost:1344/av/respmod
```

```
icap_service is_kav_req reqmod_precache 0
icap://localhost:1344/av/reqmod
icap_class ic_kav is_kav_req is_kav_resp
icap_access ic_kav allow all
```

These amendments will make the proxy transmit all requested objects to Kaspersky Anti-Virus via port 1344 of the local interface.

2.2.2. Installation on a dedicated server

Installing the application on a dedicated server is recommended where there is a high load on the proxy server, and in situations where Kaspersky Anti-Virus is used to process the traffic of several proxy servers.

Since such a deployment scheme does not allow automatic configuration of the application, it must be configured manually.

2.2.2.1. Integrating with Squid proxy server

Make the following settings to integrate the application with a stand-alone Squid proxy server:

1. After the application has been installed, edit the **ListenAddress** parameter in the **[icapserver.network]** section of the *kav4proxy.conf* configuration file to specify the IP address of the network interface and port that Kaspersky Anti-Virus will use to wait for proxy requests to process necessary objects. By default Kaspersky Anti-Virus waits for requests at the address **localhost:1344**.
2. Add these lines to the **ICAP OPTIONS** section of the proxy configuration file:

```
icap_enable on
icap_send_client_ip on
icap_service is_kav_resp respmod_precache 0
icap://<ip_address>:<port>/av/respmod
icap_service is_kav_req reqmod_precache 0
icap://<ip_address>:<port>/av/reqmod
icap_class ic_kav is_kav_req is_kav_resp
icap_access ic_kav allow all
```

where **<ip_address>** stands for the IP address of the server on which Kaspersky Anti-Virus is installed; **<port>** is the port that Kaspersky Anti-Virus uses to wait for proxy requests for anti-virus processing. Restart the proxy as soon as the changes are entered.

2.2.2.2. Integrating with other proxy servers

For comprehensive information about deploying Kaspersky Anti-Virus on the **Blue Coat ProxySG** hardware, see the **ICAP** section of the Blue Coat ProxySG Configuration and Management Guide.

For comprehensive information about deploying Kaspersky Anti-Virus on the **Cisco Content Engine** hardware see the **ICAP** section of the Cisco ACNSsoftware Command Reference, Release 5.3 - Chapter 2: Cisco ACNS Software Commands. This documentation is available online at

http://www.cisco.com/en/US/products/sw/conntsw/ps491/products_command_reference_chapter09186a00803bb081.html#wp1231699.

CHAPTER 3. INSTALLING THE APPLICATION

Before installing Kaspersky Anti-Virus, you are advised to:

1. Make sure that your system meets the hardware and software requirements (see section 1.1 on p. 6).
2. Log on to the system as **root**.

3.1. Installation on a server running Linux

Kaspersky Anti-Virus for servers running the Linux operating system is distributed in two different installation *packages*:

- *.rpm* – for systems that support RPM Package Manager.
- *.deb* – for Debian distributions.



To initiate installation of Kaspersky Anti-Virus from the rpm package, enter the following at the command line:

```
# rpm -i kav4proxy-<version_number>.rpm
```



To initiate installation of Kaspersky Anti-Virus from the deb package, enter the following at the command line:

```
# dpkg -I kav4proxy-<version_number>.deb
```

During the setup process you will have to specify additional information regarding connection to the Internet, downloading of the anti-virus databases and settings for interaction with the proxy server. Please refer to section 3.4 on p. 20 for details.

3.2. Installation on a server running FreeBSD

The distribution file for installation of Kaspersky Anti-Virus on servers running the FreeBSD operating system is supplied as a *.tgz* package.



To initiate installation of Kaspersky Anti-Virus from a tgz-package enter the following at the command line:

```
# pkg_add kav4proxy-< distributive version >.tgz
```

During the setup process you will have to specify additional information regarding connection to the Internet, downloading of the anti-virus databases and settings for interaction with the proxy server. Please refer to section 3.4 on p. 20 for details.

3.3. Installation procedure



Algorithms described in this section and in section 3.4 require that the target server already has Squid 2.5 or 3.0 with ICAP support installed.

Kaspersky Anti-Virus must be installed in two stages. The first stage will be performed automatically after execution of the commands described in sections 3.1 or 3.2, and comprises the following steps:

1. The **klusers** group and the **kluser** account are created with the necessary privileges that Kaspersky Anti-Virus will use to start and operate.
2. Copying of the files from distribution package to computer.
3. Registration of the services necessary for Kaspersky Anti-Virus to function.

3.4. Post-installation configuration

Post-installation configuration is the second stage of installation, which includes configuration of the application and of the proxy server. To initiate the configuration process, use the *postinstall.pl* script located in the folder */opt/kaspersky/kav4proxy/lib/bin/setup*. When the script starts you will be asked to perform the following actions:



The *postinstall.pl* script should be launched manually for RPM-based systems. In other systems (for example, such as FreeBSD) the script will run automatically during the installation procedure.

1. Specify the path to the license key file.
2. Configure the parameters of the proxy server which will be used to connect to the Internet, in one of these formats:

```
http://<proxy server IP address>:<port>
```

or

```
http://<user_name>:<password>@<proxy server IP address>:<port>
```

depending upon whether the proxy authenticates its users. The value will be used by the application updater component (*keepup2date*) to connect to Kaspersky Lab's update servers, when downloading updates to the anti-virus databases.

If you are not using a proxy for Internet connection, specify **no** as the value for that parameter.

3. Download updates to the anti-virus databases from Kaspersky Lab's update servers. Specify whether you wish to update immediately or later. After the update you will be asked whether to enable the automatic updates of the anti-virus databases. By default, the update is scheduled to perform hourly.
4. Configure interaction with Webmin.
5. Specify the full path to the configuration file of the Squid proxy transferring the HTTP traffic which Kaspersky Anti-Virus will scan. The settings necessary to enable interaction via ICAP between the proxy and the application will be added to this configuration file.

If you have not installed a license key during post-installation configuration, then after launch Kaspersky Anti-Virus will start functioning in **unlicensed** mode. If you have not downloaded the anti-virus databases during post-installation configuration, then after launch Kaspersky Anti-Virus will start functioning

without the anti-virus databases. Please see section 5.5 on p. 44 for details on the application modes.

3.5. Distribution of the application files in directories

The default paths for Kaspersky Anti-Virus application files on a server running Linux are:

/etc/opt/kaspersky/kav4proxy.conf – configuration file containing application parameters.

/opt/kaspersky/kav4proxy/bin – directory containing executable files of the application components:

kav4proxy-keepup2date – utility updating the anti-virus databases;

kav4proxy-licensemanager – utility for license keys management.

/opt/Kaspersky/kav4proxy/lib/bin/avbasestest – utility validating downloaded updates to the anti-virus databases used by the *keepup2date* component.

/etc/init.d/kav4proxy – application control script.

/opt/kaspersky/kav4proxy/lib/bin/setup – directory containing scripts for post-installation setup and removal of the application:

postinstall.pl – post-installation application setup script.

uninstall.pl – application removal script.

keepup2date.sh – script that configures the *keepup2date* component;

proxy_setup.pl – script that configures Squid to work with Kaspersky Anti-Virus.

/opt/kaspersky/kav4proxy/sbin/kav4proxy-kavicapserver – executable file of the main application module.

/opt/kaspersky/kav4proxy/share/doc/ – directory containing license information and deployment documentation:

LICENSE – license agreement;

README-SQUID.txt – file containing information about available Squid distributions, and correct compilation and proxy configuration for ICAP support.

/opt/kaspersky/kav4proxy/share/man – directory containing the application's manual pages.

/opt/kaspersky/kav4proxy/share/notify/ – directory containing notification templates.

/opt/kaspersky/kav4proxy/share/examples/ – directory containing examples on setting up Kaspersky Anti-Virus:

kav4proxy-default.conf – application configuration file with default settings;

notify.sh – administrator's notification script.

/var/log/kaspersky/kav4proxy/ – directory containing the application log files.

The default paths for Kaspersky Anti-Virus application files on a server running FreeBSD are:

/usr/local/etc/kaspersky/kav4proxy.conf – configuration file containing application parameters;

/usr/local/bin – directory containing executable files of the application components:

kav4proxy-keepup2date – utility updating the anti-virus databases;

kav4proxy-licensemanager – utility for license keys management.

/usr/local/libexec/kaspersky/kav4proxy/avbasestest – utility validating downloaded updates to the anti-virus databases used by the *keepup2date* component.

/usr/local/etc/rc.d/kav4proxy – application control script.

/usr/local/libexec/kaspersky/kav4proxy/setup – directory containing scripts for post-installation setup and removal of the application:

postinstall.pl – post-installation application setup script.

uninstall.pl – application removal script.

keepup2date.sh – script that configures the *keepup2date* component;

proxy_setup.pl – script that configures Squid to work with Kaspersky Anti-Virus.

/usr/local/share/kav4proxy/contrib/kav4proxy.wbm – executable file of the main application module.

/usr/local/share/doc/kav4proxy/ – directory containing license information and deployment documentation:

LICENSE – license agreement;

README-SQUID.txt – file containing information about available Squid distributions, and correct compilation and proxy configuration for ICAP support.

/usr/local/man – directory containing the application's manual pages.

/usr/local/share/kav4proxy/notify/ – directory containing notification templates.

/usr/local/share/examples/kav4proxy/ – directory containing examples on setting up Kaspersky Anti-Virus:

kav4proxy-default.conf – application configuration file with default settings;

notify.sh – administrator's notification script.

/var/log/kaspersky/kav4proxy/ – directory containing the application log files.

CHAPTER 4. USING KASPERSKY ANTI-VIRUS

This chapter describes how to carry out tasks related to the basic features of Kaspersky Anti-Virus, including updating the application, management of license keys, anti-virus protection of HTTP traffic, and configuration of anti-virus scanning parameters for different user groups. The implementation of these tasks in a specific configuration will depend upon the particular organization of the network and the existing security policy. Please refer to Chapter 5 on p. 38 for a detailed explanation of the application settings used in these tasks.

4.1. Updating the anti-virus databases

Kaspersky Anti-Virus uses the anti-virus databases while processing objects requested by users through the proxy server.

The anti-virus databases are employed while scanning for, and disinfecting, infected objects; they contain descriptions of all currently known viruses and the methods of disinfection for objects affected by those viruses.

The *keepup2date* component is included in the application to provide software updates. The updates are retrieved from the Kaspersky Lab's update servers, e.g.:

<http://downloads1.kaspersky-labs.com/>

<http://downloads2.kaspersky-labs.com/>

<ftp://downloads1.kaspersky-labs.com/> etc.

The *updcfg.xml* file included in the installation package lists the URLs of all available update servers.



The *keepup2date* component supports basic authentication for connections through a proxy server.

To update the anti-virus databases, the *keepup2date* component selects an address from the list of update servers and tries to download updates from that server. If the first server is currently unavailable, the application attempts to connect to another server, and so on until updates are downloaded or the end of the list is reached.



You are strongly advised to set up the *keepup2date* component to update the databases every hour!

After a successful update the command, specified as the value of the **PostUpdateCmd** parameter in the **[updater.options]** section of the configuration file, will be executed. By default, this command will automatically initiate the reloading of the anti-virus databases. Incorrectly modifying this parameter may prevent the application from using the updated databases, or cause it to function erroneously.



All settings of the *keepup2date* component are stored in the **[updater.*]** sections of the configuration file.

If your network has a complicated structure, you are advised to download updates from Kaspersky Lab's update servers every hour and place them in a network directory. To keep other networked computers constantly updated, configure the local computers to copy the updates from that directory. For detailed instructions on creation of a public directory, see section 4.1.3 on p. 27.

The updating process can be scheduled to run automatically using the **cron** utility (see section 4.1.1 on p. 25), or started manually from the command line (see section 4.1.2 on p. 26). Starting the *keepup2date* component requires **root** or **kluser** user privileges.

4.1.1. Automatic updating of the anti-virus databases

You can schedule regular automatic updates for the anti-virus databases using the **cron** service. You can configure *cron* either manually or using the *keepup2date.sh* script located in the */opt/kaspersky/kav4proxy/lib/bin/setup* directory.

To create a cron task which updates the anti-virus databases hourly, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/lib/bin/setup/keepup2date.sh -  
install
```

To delete this cron task, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/lib/bin/setup/keepup2date.sh -  
uninstall
```



Task: Configure the application to automatically update your anti-virus databases hourly. The system log should only record errors which occur in the component's operation. A general log should record all task starts. No information should be output to the console.



Solution: to perform the above task:

1. In the application configuration file, specify these parameter values:

```
[updater.report]
Append=true
ReportLevel=1
```

2. Edit the file that sets rules for the **cron** process (**crontab -e**) by adding the following line for the **root** or **kluser** user:

```
23 * * * *
/opt/kaspersky/kav4proxy/bin/keepup2date -q
```



The specified time setting for the **cron** task start is just an example. You are advised to specify your own settings for the start time to avoid overloading the updating servers.

4.1.2. Manual updating of the anti-virus databases

You can start an update to your anti-virus databases from the command line at any time.



Task: start updating the anti-virus databases, save the results of updating in the *updatesreport.log* file within the directory */var/log/kaspersky//kav4proxy/*.



Solution: to accomplish the task, log in as **root** (or any other privileged user) and enter at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-keepup2date
-l /var/log/kaspersky/kav4proxy/updatesreport.log
```

To update the anti-virus databases on several servers, it may be more convenient to download the updates once from an update server, save them to a shared directory, and mount the directory within the file system of the other servers running Kaspersky Anti-Virus. Then it will be sufficient to launch the update script on these other servers, having first specified the mounted directory as the source of updates. FTP and HTTP can also be used to share the anti-

virus databases. Please see section 4.1.3 on p. 27 for details related to creation of a shared directory for updates.



Task: initiate updating of the anti-virus databases using the local `/home/kluser/bases` directory as the source. Output the results to the `/tmp/updatesreport.log` file.



Solution: to accomplish the task, log in as **root** (or any other privileged user) and perform the following steps:

1. Mount the shared directory containing updates to the anti-virus databases as the local `/home/kluser/bases` directory.
2. Enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-keepup2date -l /tmp/updatesreport.log -g /home/kluser/bases
```



You can also update the application remotely using the appropriate Webmin plug-in.

4.1.3. Creating a shared directory for storing and sharing database updates

To correctly update the anti-virus databases on local computers from a shared directory, that directory must have the same file system structure as Kaspersky Lab's update servers.



Task: create a shared local directory which local computers will use as the source of anti-virus database updates.



Solution: to accomplish the task, log in as **root** (or any other privileged user) and do the following:

1. Create a local directory. The **kluser** account must have sufficient privileges to write to it.
2. Run the `keepup2date` component as follows:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-keepup2date -u <dir>
```

where `<dir>` stands for the full path to the created directory.

3. Provide reading access to that directory for local computers on your network.

4.2. Managing license keys

The right to use Kaspersky Anti-Virus is determined by the *license key*. The key is included in the product's distribution kit, and entitles you to use the application as soon as you purchase it. It contains all the information related to the license you have purchased, including the type of license, the license's expiry date, and information about dealers.



The application checks for the presence of an installed license key every time it starts or reloads its anti-virus databases.

If a license key is not installed, or an error has occurred while loading information about the current license, the application switches into a special *unlicensed* mode of operation. In that mode it does not perform anti-virus scanning of objects transferred through the proxy server; instead, all objects are treated using the action specified by the **LicenseErrorAction** parameter (see section A.1 on p. 55).

After the license expires, the functionality of the application will still be preserved except for the ability to update the anti-virus databases. You will still be able to perform anti-virus scanning and processing of objects, but you will be unable to use databases issued after the license expiration date. Therefore, you may not be protected against new viruses that have appeared after the license expired.

To protect your computer against new viruses, you are advised to renew the license.

In addition to the right to use the application during the license period, the license gives the following benefits:

- twenty-four-hour technical support
- hourly updates of the anti-virus databases
- timely notifications about new virus threats.

Therefore it is essential to extend your license to use Kaspersky Anti-Virus in a timely fashion. You can also install an additional key, which the application will start using as soon as the current active key expires (see section 4.2.2 on p. 31).

4.2.1. Viewing information about license keys

You can view information about installed license keys in the reports of the *kavicapserver* component. Each time the component starts, *kavicapserver* loads the license key information and displays it in the report. The *kavicapserver.log* report file is stored in the */var/log/kaspersky/kav4proxy/* directory.

More detailed information about the status of license keys may be obtained using *licensemanager*, a special component of the application.

All information about license keys may be viewed either on the server's console, or remotely from any networked computer that has access to the Webmin module.



To view information about all installed license keys, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -s
```

In the server console, you will see information similar to the following:

```
Kaspersky license manager for Linux. Version  
5.5.0/RELEASE
```

```
Copyright (C) Kaspersky Lab, 1997-2006.
```

```
Portions Copyright (C) Lan Crypto
```

```
License info:
```

```
Product name: Kaspersky Anti-Virus for Proxy Server
```

```
Expiration date: 14-06-2006, expires in 117 days
```

```
Active key info:
```

```
Product name: Kaspersky Anti-Virus for Proxy Server
```

```
Key file      0009A3A3.key
```

```
Type:        Commercial
```

```
Expiration date: 14-06-2006
```

```
Serial:      0007-00047E-0009A3A
```



To view information about a license key, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -k <keyfilename>
```

where <keyfilename> is the name of the license key file, for instance 00053E3D.key.

In the server console, you will see information similar to the following:

```
Kaspersky license manager. Version 5.5.0/RELEASE  
Copyright (C) Kaspersky Lab, 1997-2006.  
Portions Copyright (C) Lan Crypto  
Product name:   Kaspersky Anti-Virus for Proxy Server  
Creation date:  15-03-2005  
Expiration date: 14-06-2006  
Serial          0007-00047E-0009A3A  
Type:           Commercial  
Count:          1  
Lifespan:       365
```



To view detailed information about the licensing parameter, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -i
```

In the server console, you will see information depending on the licensing type. For example, in case of the traffic amount-based license:

```
Kaspersky license manager for Linux. Version  
5.5.31/RELEASE #17  
Copyright (C) Kaspersky Lab, 1997-2007.  
Portions Copyright (C) Lan Crypto  
  
Licensed traffic units:   500 (MB)  
Traffic units used: 0(MB)  
Traffic units left: 500 (MB)
```

4.2.2. Renewing your license

Renewing the Kaspersky Anti-Virus license will give you the right to re-enable full product functionality, and resume the use of the additional services listed in section 4.2 on p. 28.

The license term depends on the product you bought and the type of the license you purchased.



To renew the license for Kaspersky Anti-Virus:

Contact the company that sold you the product, and renew your license for Kaspersky Anti-Virus.

or:

Purchase a license extension directly from Kaspersky Lab. Write a letter of request to our Sales Department at sales@kaspersky.com, or fill in the corresponding form on our website www.kaspersky.com in the section **E-Store → Renew Your License**. After your payment is received, we will send a license key to the email address indicated in the corresponding field of your license renewal form.



To install a new license key, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -a <keyfilename>
```

where <keyfilename> is the name of the license key file, for instance 00053E3D.key.

If installation is successful, information similar to the following will be displayed on the server console:

```
Kaspersky license manager. Version 5.5.0/RELEASE  
Copyright (C) Kaspersky Lab. 1998-2006.  
Key file 00053E3D.key is successfully registered
```

We recommend that you update the anti-virus databases after the installation.

If you want to install a new license key before the current license key expires, you can add it as a backup license key. The backup key will be activated immediately the current one expires. The term of validity for the additional key starts from the activation date. You can install only one backup key.

If you have installed two keys (the current and an additional one), you can view information about both of them in the server console.

4.2.3. Removing a license key



To remove the current license key, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -da
```

If the component removes the license key successfully, information similar to the following will be displayed on the server console:

```
Kaspersky license manager. Version 5.5.0/RELEASE  
Copyright (C) Kaspersky Lab. 1998-2006.  
Active key was successfully removed
```



To remove a backup key, enter the following at the command line:

```
# /opt/kaspersky/kav4proxy/bin/kav4proxy-  
licensemanager -dr
```

The server console will display information similar to the following:

```
Kaspersky license manager. Version 5.5.0/RELEASE  
Copyright (C) Kaspersky Lab. 1998-2006.  
Additional key was successfully removed
```

4.3. Using a control script

The *kav4proxy* application control script located in the */etc/init.d* directory is used to start, stop and restart the application. It uses the following command line parameters:

- **start** – command to check the configuration file and launch the application. A return code of **0** indicates a successful start.
- **stop** – command to stop the application. If the application is running, the script sends the SIGTERM signal. If the application does not stop within 30 seconds, the script sends the SIGKILL signal. A return code of **0** indicates a successful execution.

- **restart** – command to stop and restart the application, as provided by using the **stop**, and then **start**, keys.
- **reload** – command to reload the application configuration and the anti-virus databases using the SIGHUP signal.
- **reload_avbase** – command to reload only the anti-virus databases, and validate the license key.
- **stats** – command to write the results of statistics counters to a file (see section 5.7 on p. 46) and switch the report logging to another file. This command line option can be used in systems which automatically rotate log files, to initiate logging into a new file.

4.4. Ensuring anti-virus protection of HTTP traffic



Task:

Provide anti-virus scanning of HTTP traffic transmitted by a proxy server installed on the same server as Kaspersky Anti-Virus, in accordance with the following requirements:

- General parameters of anti-virus scanning must be used for all requests:
 - Disinfection of infected objects must be enabled.
 - Scanning of e-mail databases must be disabled.
 - Scanning of packed and archived objects must be enabled.
- Block access to infected, suspicious and damaged objects, and objects containing code that resembles a known virus.
- Use **partial** mode (please refer to 5.6 on p. 45) while processing proxy server requests.
- Disable anti-virus scanning of objects requested from the *www.trusted_company.com* web server.
- Store statistics on the results of anti-virus scanning in the */var/log/kaspersky/kav4proxy/statistic* file.



Solution: to accomplish the task, perform these steps:

1. Install Kaspersky Anti-Virus on the same server as the proxy server (see section 2.2.1 on p. 15) and perform its post-installation configuration (see section 3.4 on p. 20).
2. Specify the following parameter values in the *kav4proxy.conf* configuration file (leave the values of parameters not mentioned in the example unchanged):

```
[icapservice.filter]
ExcludeURL==^www\.trusted-company.com\/.*

[icapservice.engine.options]
ScanPacked=yes
ScanArchives=yes
ScanMailBases=no
ScanMailPlain=no
Cure=yes

[icapservice.actions]
InfectedAction=deny
SuspiciousAction=deny
WarningAction=deny
ErrorAction=skip
ProtectedAction=skip
CorruptedAction=skip

[icapservice.protocol]
AnswerMode=partial

[icapservice.statistics]
AVStatisticsFile=/var/log/kaspersky/kav4proxy/statistic
```

3. Restart Kaspersky Anti-Virus using the following command:

```
# /etc/init.d/kav4proxy reload
```

Please refer to Chapter 5 on p. 38 for a more detailed description of the settings used in the solution for the task.

4.5. Configuring anti-virus scan parameters for user groups

The example in section 4.4 uses common settings for anti-virus processing of all user requests coming through the proxy server. Kaspersky Anti-Virus allows the definition of groups, to allow different parameters to be used for anti-virus protection of individual users.



Task:

Configure the application to perform anti-virus checks of HTTP traffic in accordance with the following requirements:

- These anti-virus scanning parameters must be specified for the **managers** group, which comprises computers using IP addresses on the 192.168.1.0/255.255.255.0 subnet:
 - Scanning of packed, archived files and e-mail databases must be disabled.
 - Disinfection of infected objects must be enabled.
 - Access should be granted to clean and disinfected objects only.
- These anti-virus scanning parameters must be specified for the **sales** group, which comprises computers using IP addresses on the 192.168.2.0/255.255.255.0 subnet:
 - Scan all objects.
 - Disinfection of infected objects must be enabled.
 - Block access to infected, suspicious and damaged objects, and objects containing code that resembles a known virus.
- These anti-virus scanning parameters must be specified for all other users:
 - Scanning of e-mail databases must be disabled.
 - Disinfection of infected objects must be disabled.
 - Access should only be granted to objects that have been assigned the **OK** status after a scan (please refer to section 5.3 on p. 40 for details about statuses).



Solution: to accomplish the task, perform the following steps:

1. In the *kav4proxy.conf* configuration file, create the following sections containing anti-virus scanning parameters for the **managers** group:

```
[icapservice.groups:managers]
Priority=1
ClientIP=192.168.1.0/255.255.255.0
URL=.*
```

```
[icapservice.engine.options:managers]
ScanPacked=no
ScanArchives=no
ScanMailBases=no
ScanMailPlain=no
Cure=yes
```

```
[icapservice.actions:managers]
InfectedAction=deny
SuspiciousAction=deny
WarningAction=deny
ErrorAction=deny
ProtectedAction=deny
CorruptedAction=deny
```

2. In the *kav4proxy.conf* configuration file, create the following sections containing anti-virus scanning parameters for the **sales** group:

```
[icapservice.groups:sales]
Priority=2
ClientIP=192.168.2.0/255.255.255.0
URL=.*
[icapservice.engine.options:managers]
ScanPacked=yes
ScanArchives=yes
ScanMailBases=yes
```

```
ScanMailPlain=yes  
Cure=yes
```

```
[icapserver.actions:sales]  
InfectedAction=deny  
SuspiciousAction=deny  
WarningAction=deny  
ErrorAction=skip  
ProtectedAction=skip  
CorruptedAction=deny
```

3. Specify the following parameters for the default group:

```
[icapserver.engine.options]  
ScanPacked=yes  
ScanArchives=yes  
ScanMailBases=no  
ScanMailPlain=no  
Cure=no
```

```
[icapserver.actions]  
InfectedAction=deny  
SuspiciousAction=deny  
WarningAction=deny  
ErrorAction=deny  
ProtectedAction=deny  
CorruptedAction=deny
```

4. Restart Kaspersky Anti-Virus using the following command:

```
# /etc/init.d/kav4proxy reload
```

Please refer to Chapter 5 on p. 38 for a more detailed description of the settings used in the solution for the task.

CHAPTER 5. DETAILED SETTINGS FOR KASPERSKY ANTI-VIRUS

This chapter contains a detailed explanation of basic parameters of Kaspersky Anti-Virus. Unlike the required settings essential for application functioning, which are specified during installation and post-installation configuration, additional configuration can be performed at the administrator's discretion. It is intended to extend the application's functionality, and its ability to enforce your corporate security policy.

5.1. Creating groups

The use of groups allows an administrator to specify different anti-virus processing for objects being requested or transferred through a proxy server by different user groups. A request is associated with a specific group depending on the IP address of the client computer requesting the object through a proxy server, and the URL of that object.



Ensure that the **icap_send_client_ip** parameter in Squid configuration file is set to **on**. This value means that Squid will transfer the client's IP address to Kaspersky Anti-Virus.

If a request's parameters do not match any existing group, the application will process the requested objects in accordance with the rules specified for the default group.

Each group's parameters are stored in the following five sections of the application's configuration file:

- **[icapserver.groups:<group name>]** contains parameters that define the group applicability range (IP addresses of clients, object URLs) and the group's priority.
- **[icapserver.filter:<group name>]** contains filtration rules for the **<group name>** group.
- **[icapserver.engine.options:<group name>]** contains anti-virus scanning parameters used to process objects associated with the group.

- **[icapservr.actions:<group name>]** contains parameters that determine what actions are performed by the application on objects with a particular anti-virus scan status.
- **[icapservr.notify:<group name>]** contains parameters used in notifying administrators about blocked objects, to which the application has applied the **deny** action.



The default group parameters are specified in the **[icapservr.groups]**, **[icapservr.filter]**, **[icapservr.options]**, **[icapservr.actions]** and **[icapservr.notify]** sections.

You do not have to specify all group parameters while creating a new group. If some parameters are missing, the application uses the default settings (see Appendix A.1 on p. 55).



Example: create the **managers** group to define rules for processing objects requested by client computers using the subnet 192.168.10.0/255.255.255.0. Prevent the group from accessing any objects that are not clean, disinfected and password-protected. Set the group priority to **2**. Use default values for all other parameters.



Solution: to accomplish the task, log in as the **root** (or any other privileged user) and create these sections in the *kav4proxy.conf* configuration file:

```
[icapservr.groups:managers]
Priority=2
ClientIP=192.168.10.0/255.255.255.0
URL=.*
```

```
[icapservr.engine.options:managers]
Cure=yes
```

```
[icapservr.actions:managers]
CuredAction=skip
ErrorAction=deny
ProtectedAction=skip
```

5.2. Anti-virus scan settings

The anti-virus engine parameters in the [**icapserver.engine.options:<group name>**] section define modes for scanning and disinfecting requested objects within a corresponding group, as follows:

- **ScanPacked=yes|no** – enables/ disables scanning of packed files. If the mode is disabled, all packed objects are considered to be clean.
- **ScanArchives=yes|no** – enables/ disables scanning of objects inside archives. If the mode is disabled, all archive files are considered to be clean.
- **ScanMailBases=yes|no** – enables/ disables scanning of email databases (either requested or transferred via a proxy server). If the mode is disabled, all email databases are considered to be clean.
- **ScanMailPlain=yes|no** – enables/ disables scanning of email messages in *plain text* format (requested or transferred via a proxy server). If the mode is disabled, all plain text email is considered to be clean.
- **UseHeuristic=yes|no** – enables/ disables heuristic analyzer used for anti-virus scanning.
- **Cure=yes|no** – enables/ disables disinfection of infected objects. If the mode is disabled, the application will not attempt to cure an infected file.
- **UseAVbasesSet=standard|extended** – the set of anti-virus databases to be used by the application. The **extended** set contains, in addition to the records of the **standard** set, the signatures of other potentially dangerous software such as adware and remote administration utilities.

5.3. Choosing actions for scanned objects

Actions performed by the application on scanned objects are defined by the status assigned to those objects following an anti-virus check.

Kaspersky Anti-Virus uses the following statuses:

- **OK** – clean object that has successfully passed the scanning procedure and is not infected.
- **INFECTED** – the object is infected; either it cannot be cured, or disinfection has not been attempted.

- **CURED** – the object was infected, but has been cured successfully.
- **WARNING** – the object contains code that resembles a known virus.
- **SUSPICIOUS** – the object is suspected of being infected with an unknown virus.
- **PROTECTED** – the object is password-protected and therefore cannot be scanned.
- **CORRUPTED** – the object is damaged.
- **ERROR** – object scanning resulted in an error.

Actions performed by Kaspersky Anti-Virus on objects with a specific status are determined by the parameters in the **[icapserv.actions]** section (for the default group) and **[icapserv.actions:<group name>]** section (for groups created by the administrator):

- **InfectedAction** – action taken on infected objects that have not been cured or cannot be cured.
- **SuspiciousAction** – action taken on objects suspected of being infected with an unknown virus.
- **WarningAction** – action taken on objects containing code that resembles a known virus.
- **ErrorAction** – action taken on objects that have been assigned the **ERROR** status.
- **ProtectedAction** – action taken on password-protected objects.
- **CorruptedAction** – action taken on damaged objects.
- **CuredAction** – action taken on disinfected objects.

The parameters defining these action can take the following values:

- **skip** – allows object transfer.
- **deny** – prohibits object transfer, replacing the object with a corresponding notification file.

If **deny** is the action used on an object, then, depending upon the object's status, it will be replaced with one of the following placeholder files located in the */opt/kaspersky/kav4proxy/share/notify:* directory:

- **object_infected** – template containing a notification about detection of an infected object.
- **object_suspicious** – template containing a notification about detection of an object suspected of being infected with an unknown virus.

- **object_warning** – template containing a notification about detection of an object that resembles the code of a known virus.
- **object_protected** – template containing a notification about detection of a password-protected object.
- **object_error** – template containing a notification about detection of an object which caused a scanning error.
- **object_corrupted** – template containing a notification about detection of a damaged object.
- **object_cured** – template containing a notification about detection of an infected object that has been successfully cured.

Administrators can modify the text of these templates at their discretion, including addition of special macros (see Appendix A.2 on p. 62).



Example: Specify the following actions for scanned objects for the default group:

- allow transfer of the objects that have been assigned the **CURED** and **PROTECTED** status
- prohibit transfer of all other objects.



Solution: to accomplish the task, log in as **root** (or any other privileged user) and specify the following parameter values in the **[icapservr.actions]** section:

```
[icapservr.actions]
CuredAction=skip
ProtectedAction=skip
InfectedAction=deny
SuspiciousAction=deny
WarningAction=deny
ErrorAction=deny
CorruptedAction=deny
```

5.4. Administrator notifications

Every time the application performs the **deny** action on an object transferred through the proxy, it also runs a special script. Such script's example is located at: `/opt/kaspersky/kav4proxy/share/examples/notify.sh`. The **NotifyScript** parameter, in the **[icapservr.notify:<group name>]** section of the application configuration file, contains the script's filename.

Below you can examine a sample notification script and the steps necessary to configure the application to run the script.



Administrators can use SHELL syntax to create their own custom scripts, which will be executed every time the application blocks an object transfer via the proxy after scanning it. Every group created by the administrator can be assigned its own notification script (please refer to section 5.1 on p. 38 for details about groups).



To configure the application to send notifications about blocked objects to `admin@test.local`, perform the following steps

1. Create an executable script file with the following contents:

```
#!/bin/sh
recipients='admin@test.local'

/usr/lib/sendmail -t -i<<EOT
From: Kaspersky Anti-Virus For Proxy Server
<root@$HOSTNAME>
To: $recipients
Subject: %VERDICT% object requested

Action applied: %ACTION%
Verdict: %VERDICT%
Requested URL: %URL%
Client IP: %CLIENT_ADDR%

Found:
    Infected: %VIRUS_LIST%
    Cured: %CURED_LIST%
    Suspicious: %SUSP_LIST%
    Warnings: %WARN_LIST%

This message generated by %PRODUCT% at %DATE% on
$HOSTNAME
EOT
```



During script creation you can use special macros, such as `%URL%`, `%CLIENT_ADDR%`, etc. to specify additional information. Please refer to section A.2 on p. 62 for details on macros.

2. Save the script file and make sure that the **kluser** user account has sufficient privileges for its execution.
3. Set the script's filename as the value of the **NotifyScript** parameter. For instance, if the script has been saved as the file `/usr/local/bin/notify.sh`, and it should be executed whenever objects processed according to the default group rules are blocked, specify the following value for the **NotifyScript** parameter in the **[icapserver.notify]** section:

```
[icapserver.notify]
NotifyScript=/usr/local/bin/notify.sh
```



The application installation contains notification templates, which can be used when creating scripts. By default, these templates are located at `/opt/kaspersky/kav4proxy/share/notify`.

5.5. Operation modes

Depending on the status of the license and of the anti-virus databases, the application can function in one of the following modes:

- **Basic mode** – fully functional mode of application operation. In this mode the application performs anti-virus scanning of proxy traffic, and disinfection of infected objects (if enabled).
- **Operation without updates** – the mode used by the application when the current license expires. In this mode the application performs anti-virus scanning of proxy traffic and, if enabled, disinfection of infected objects using the anti-virus databases current at the moment of license expiry.
- **Unlicensed operation** – the mode used by the application when the license key is not installed, or when an error has occurred while loading the information about the current license. In this situation the application does not perform anti-virus scanning of proxy traffic, and applies to all objects the action defined by the **LicenseErrorAction** parameter.
- **Operation without the anti-virus databases** – the mode used by the application if its anti-virus databases are not installed or if an error has occurred while loading them. In this mode the application does not

perform anti-virus scanning of proxy traffic, and applies to all objects the action defined by the **BasesErrorAction** parameter.

5.6. Modes of interaction with proxy via ICAP

The mode used by Kaspersky Anti-Virus to work with a proxy server is defined by the **AnswerMode** parameter in the **[icapserver.protocol]** section of the *kav4proxy.conf* configuration file, which can take the following values:

- **partial** – in this mode, Kaspersky Anti-Virus sends parts of the object being scanned to the proxy server, with the frequency determined by the **MaxSendDelayTime** parameter for their further transfer to the user. The last part of an object will only be sent to the user when the anti-virus scan of the object is complete, and only if the resulting status does not mean that the **deny** action should be applied to that object. If the **deny** action is applied to the object, the application does not send a template-based file to the user (see section 5.3 on p. 40); instead, the application will initiate disconnection.



This mode is convenient when large files are downloaded. In this case, users begin receiving objects before completion of an anti-virus check: otherwise, a user may terminate connection before he/she receives a response because of a long waiting period.

- **complete** – in this mode, Kaspersky Anti-Virus returns an object to the proxy server only after it is downloaded and tested completely, and provided that its resultant status does not require the **deny** action. If the **deny** action is applied to the object because of its status, the application will return a template-based file to the user, instead of the requested object (see section 5.3 on p. 40).



When **complete** mode is used, after clicking on an object in the browser window, the user will not see a dialog box allowing him/her to save the object or cancel scanning until the object is completely downloaded by the proxy server and scanned by Kaspersky Anti-Virus. The download can only be cancelled by closing the browser window, thus terminating the connection.

5.7. Logging application statistics

Kaspersky Anti-Virus provides two types of statistical information for administrators:

- Statistics on the results of anti-virus scanning and processing
- General statistics on the application's activity.

Statistics of anti-virus processing can be written to a local file or to a network socket. To log statistics to a local file, specify the path to that file as the value for the **AVStatisticsFile** parameter. The **AVStatisticsAddress** parameter is intended to specify a network socket.

Every line in the resulting statistics file will contain information about a single tested object, in the following format:

```
Time Size Verdict Virus_info IP URL
```

Table 2 contains a summary of these parameters.

Table 2. Statistics parameters

Alias	Meaning
Time	Date of object scanning.
Size	Object size.
Verdict	Status assigned to the object after anti-virus scan.
Virus_info	List of revealed viruses.
IP	IP address of the client that requested the object.
URL	URL of the requested object.

In addition to the statistics of anti-virus scanning, the application also uses special counters which return statistical information about its activity. These counter values can be output to a file specified in the **CounterStatisticsFile** parameter in the application configuration file. The resulting file will contain a log of values returned by counters, as described in Table 3.

Table 3. Counters of application activity

Counter	Description
Total_requests	Total number of processed scan requests.
Infected_requests	The number of requests which returned infected or suspicious objects, or objects resembling a known virus.
Protected_requests	The number of requests which returned protected objects.
Error_requests	The number of requests which returned objects causing processing errors.
Processed_traffic	The total volume of processed traffic, including service traffic (MB).
Clean_traffic	The total volume of clean traffic (MB).
Infected_traffic	The total volume of infected traffic (MB).
Traffic_per_min	Average MB per minute.
Request_per_min	Average number of ICAP requests processed per minute.
Total_connections	The number of active connections to the ICAP server.
Total_processes	The total number of running processes working on user requests.
Idle_processes	The number of idle processes waiting for requests.

5.8. Application reporting parameters

The results of operations performed by components of Kaspersky Anti-Virus are summarized in a log file in text format, specified by the **ReportFileName** parameter in the **[icapsver.report]** section. If an empty string is set as the value of the **ReportFileName** parameter (**ReportFileName=**), no information about application activity will be logged.

The amount of output information can be altered by changing the *report detail level*, set by the **ReportLevel** parameter in the **[icapsver.report]** section.

The **level of detail** is a number that sets the level of verbosity for information regarding the components' work. Each subsequent level includes information of the previous level together with some additional data.

Possible levels of report details are listed in the table 4 below.

Table 4. Levels of report details

Level	Level name	Level letter symbol	Meaning
0	Fatal Errors	F	Information about critical errors only (i.e. errors which cause program termination because some actions cannot be performed). For instance, virus infection of a component, or an error while initializing or loading databases and license keys.
1	Errors	E	Information about other errors which do not cause termination of components' activity; for example, information about an error encountered during file scanning.
2	Warning	W	Notifications about errors that may lead to the application shutdown (license key expiration warning, out-of-disk-space warning, etc.).

Level	Level name	Level letter symbol	Meaning
3	Info, Notice	I	Important informational messages, such as whether a component is running or inactive, the path to the configuration file, the scan scope, database updates, license keys, statistics summary.
4	Activity	A	Messages about scanning of files in accordance with the level of details defined for the report.
9	Debug	D	All debug messages.

Information about fatal errors is always displayed, regardless of the report detail level. The optimal level is level **4**, which is also the default level.

Information messages may be subdivided into the following types:

- Messages pertaining to anti-virus checks.
- Messages pertaining to the operation of the application.

The output format for each of the detail levels listed above is as follows:

```
[DD-MM-YY HH:MM:SS L] STRING
```

where

DD-MM-YY HH:MM:SS stand for the date and time of record creation in the format defined by the **DateFormat** and **TimeFormat** parameters.

L – letter symbol indicating the selected level of details in report.

STRING – text containing information about the event.

For example, information about the results of anti-virus scan for an object will be logged in the following format:

```
[DD-MM-YY HH:MM:SS A] CLIENT_IP URL VERDICT [INFO]
```

where

DD-MM-YY HH:MM:SS have the same meanings as for the general message above.

A means the Activity level.

CLIENT_IP – IP address of the client that has requested on object.

URL – URL of the requested object.

VERDICT – object status determined by its anti-virus scanning.

INFO – additional information, for instance, the name of a detected virus.

5.9. Creating a memory dump to detect errors

Memory dump files or *core files* are created during an emergency shutdown of the application process; they can be used later by experts at Kaspersky Lab to identify the cause of problems in the operation of Kaspersky Anti-Virus.

The creation of core files is disabled by default: it is only recommended for the detection of problems which cause abnormal termination of application processes.

To enable the creation of memory dump files, specify the path `/var/log/kaspersky/kav4proxy/core` as the value of the **CorePath** parameter in the `[icapservers.path]` of the application configuration file.



Make sure that the partition where the `/var/log/kaspersky/kav4proxy/core` directory is located has sufficient free disk space for storage of core files.

In addition, in FreeBSD-based systems a modification of system kernel parameters may be necessary, which can be performed by running the following command as **root**:

```
# sysctl -w kern.sugid_coredump=1
```

Now in the case of an emergency shutdown of the application, a file containing a dump of its memory will be created in the `/var/log/kaspersky/kav4proxy/core` directory.

As soon as the core files have been used to collect the necessary information, you are advised to disable their creation and roll back the modifications (if any) to the system kernel in FreeBSD-based systems by running this command as **root**:

```
# sysctl -w kern.sugid_coredump=0
```

5.10. Work with Internet broadcasting stations

If Kaspersky Anti-Virus is used to anti-virus process web traffic generated by Internet radio stations, it can interrupt the data stream transfer or the proxy server operation which makes listening to broadcasts via a proxy a bit complicated. In such cases you are advised to exclude such traffic from the scope of anti-virus scanning using the **ExcludeMimeType** parameter, for example:

```
[icapserver.filter]
ExcludeMimeType=^audio/mpeg$
ExcludeMimeType=^application/vnd.ms.wms-hdr.asfv1$
ExcludeMimeType=^application/x-mms-framed$
```

These settings will exclude data streams in MPEG, ASF and Microsoft Windows Media formats from the scope of anti-virus scanning. Add to these excluded MIME types the format used by the Internet radio station that you would like to listen to.

5.11. Optimizing Kaspersky Anti-Virus

Kaspersky Anti-Virus can be optimized to reduce response time and traffic. Two major reasons for reduced performance are:

- sending large amounts of data between Kaspersky Anti-Virus and the proxy server
- scanning all objects, without distinction.

The application supports the **204 No Content** response. Using this feature helps to reduce traffic.

Scanning all objects without distinction can be avoided by setting up exclusions.

The rest of this section discusses these optimizations.

5.11.1. Reducing traffic

In some cases, an object received from a proxy server is not modified by the application (for example, when the object is not infected). If the application is

functioning in “complete” mode (see 5.6 on p. 45), the entire object will be sent back to the proxy server.

If the application is interacting with a proxy server in “partial” mode (see 5.6 on p. 45) and the checked object is small, the application can complete scan before the **MaxSendDelayTime** period (see 5.6 on p. 45) expires. In this situation also the entire object will be sent to the proxy server.

Use the **204 No Content** response to avoid unnecessary traffic. Assign the value **true** to the **Allow204** parameter in the **[icapserver.protocol]** section of the application configuration file. After that the **204** response is used instead of sending the entire object.

5.11.2. Setting up exclusions

One way to improve Kaspersky Anti-Virus performance is to set up exclusions. There are three types of exclusion rules:

- exclusion by the object’s URL
- exclusion by the object’s type
- exclusion by the object’s size.

When excluding objects by their URLs, the application compares the object’s URL with the **ExcludeURL** parameter value in the **[icapserver.filter]** section of the application configuration file. If the comparison succeeds, no virus scan is performed and the **204 No Content** response is sent to the proxy server.

When excluding objects by the object’s type, the application analyses the **Content-Type** field from the HTTP header of the received object. If the **Content-Type** matches one of the **ExcludeMimeType** parameter values in the **[icapserver.filter]** section of the *kav4proxy.conf* file, no virus scan is performed and the **204 No Content** response is sent to the proxy server.

When excluding objects based on their size, the application checks the **Content-Length** field from the object’s HTTP header. If this field’s value exceeds the **MaxReqLength** parameter value from the **[icapserver.filter]** section of the *kav4proxy.conf* file, no virus scan is performed and the **204 No Content** response is sent to a proxy.

Enable the ICAP **preview** feature to use exclusions more effectively. The **preview** feature allows the initial part of the object, containing the HTTP header, to be received, instead of the whole. The header contains enough information to apply the exclusion rules. If a rule is applied, the **204 No Content** response is sent to the proxy server, decreasing traffic between the proxy and Kaspersky Anti-Virus.

The size of the initial part of the received object is specified via the **PreviewSize** parameter in the **[icapservr.protocol]** section of the *kav4proxy.conf* file. The proxy server must be properly configured to enable preview. For Squid proxy servers, the preview feature is enabled via the **icap_preview_enable** parameter of the Squid configuration file.

CHAPTER 6. UNINSTALLING THE APPLICATION

Depending upon which distribution package was used to install the application, removing Kaspersky Anti-Virus from a server running Linux will require one of these steps:

- To uninstall the application installed from an .rpm package, type the following at the command line:

```
# rpm -e <distribution_package_name>
```

- To uninstall the application installed from a .deb package, type the following at the command line:

```
# dpkg -r <distribution_package_name>
```

To remove Kaspersky Anti-Virus from a server running FreeBSD, type the following at the command line:

```
# pkg_delete <distribution_package_name>
```

The procedure removing Kaspersky Anti-Virus runs automatically; it successively performs these operations:

1. Removes the cron task updating the anti-virus databases from the list of tasks for the **kluser** user.
2. Removes settings, made by the application in the Squid proxy server configuration file and restarts the proxy server.
3. Terminates the application services.
4. Rolls-back the registration for automatic start-up of application services in the system.
5. Removes temporary files and directories created while Kaspersky Anti-Virus was running.
6. Removes application files: the procedure deletes all the application's directories and files, including the anti-virus databases installed with the package. The only exceptions are reports, configuration files and the backup directory, which will not be deleted.

APPENDIX A. APPLICATION REFERENCE

A.1. *kav4proxy.conf* application configuration file

This appendix gives a detailed description of the *kav4proxy.conf* file, which contains all settings for Kaspersky Anti-Virus. Immediately after installation, parameters are set to the application's default settings.

The **[path]** section contains parameters defining paths to the directories essential for the application's functioning:

BasesPath – full path to the directory where the anti-virus databases are stored.

LicensePath – full path to the directory where the license keys for the application are stored.

TempPath – full path to the directory where the application's temporary files are stored.

The **[options]** section contains the parameters that define the user and the group used to run the application:

User – name of the user whose privileges the application uses to run (**kluser** by default).

Group – name of the group whose privileges the application uses to run (**klusers** by default).

The **[locale]** section contains the parameters that define the date and time format in reports and application statistics:

DateFormat=%d-%m-%Y – date format in the application activity report.

TimeFormat=%H:%M:%S – time format in the report.

The **[icapservers.network]** section contains network settings of the application:

ListenAddress – IP address and the port that Kaspersky Anti-Virus uses to wait for proxy requests sent via ICAP. Default value: **localhost:1344**.

Timeout – network timeout for interaction via ICAP.

The **[icapserver.process]** section contains advanced settings for Kaspersky Anti-Virus processes:

MaxChildren – the maximum number of running child processes working on requests sent via ICAP.

IdleChildren – the maximum number of running child processes waiting for requests sent via ICAP.

MaxReqsPerChild – the maximum number of requests to scan objects that a child process can serve. After processing the specified number of requests, a child process terminates and the application initiates a new child process.

MaxEnginesPerChild – the maximum number of scanning modules used simultaneously by child processes for anti-virus scanning of objects.



A larger number of scanning modules allows faster anti-virus scanning of objects, at the expense of server's other performance (for instance, GUI interaction). Please take into account the hardware of your server when setting this parameter's value. You can optimize Kaspersky Anti-Virus' performance as described in section 5.11 on p. 51.

The **[icapserver.protocol]** section contains the settings for the interaction between Kaspersky Anti-Virus and the proxy server via ICAP:

AnswerMode=partial|complete – the method of interaction with the proxy. The **partial** value means that Kaspersky Anti-Virus will allow transfer of the scanned object's parts to the client before the object is completely downloaded from the Internet and scanned. The **complete** value means that Kaspersky Anti-Virus will only allow transfer of a requested object to the client after it is downloaded completely and scanned.

MaxSendDelayTime – time interval (seconds) that determines the frequency used to send parts of a requested object to the client in **partial** mode.

HTTPClientIpField – name of the HTTP field to be used to identify the client's IP address.

PreviewSize – the size of **preview** request. If the parameter value is 0, then the server refuses to receive preview requests.

MaxConnections – the maximum number of connections allowed for the current ICAP server. This parameter's value is returned to the ICAP

client via the OPTIONS method. If the parameter value is 0, then the OPTIONS method does not return the number of connections.

Allow204 – allows/prohibits using of the standard ICAP response **204 No Content**. The default value is **true**.

The **[icapserver.statistics]** section contains the parameters pertaining to the generation of application statistics:

CounterStatisticsFile – path to the file where the values of statistics counters will be stored.

AVStatisticsFile – path to the file where anti-virus scanning statistics will be stored.

AVStatisticsAddress – network socket for logging anti-virus scanning statistics.

The **[icapserver.report]** section contains the parameters pertaining to report generation by the application:

ReportFileName – filename for the report on application activity.

Buffered=yes|no – buffer mode for recording to the report file. To disable the mode, set **no** as the parameter value.

ReportLevel – level of details in report.

ShowOk=yes|no – the logging mode for information about objects where scanning revealed no malicious code.

Append=yes|no – report generation mode in which the report is created anew each time the application starts. If you wish to add new information to an existing report instead of overwriting it, set the parameter value to **yes**.



Sections described above contain the anti-virus processing parameters for the default group. Please refer to section 5.1 on p. 38 for details about creation of your own groups.

The **[icapserver.path]** section contains parameters that define the paths to special application files:

PidFile – path to the PID file of the application. Default value: **/var/run/kavicapserver.pid**.

CorePath – directory in which to store memory dump files (core files) created in case of emergency termination of the application. The default empty value disables creation of core files. To enable memory dump

creation, specify `/var/log/kaspersky/kav4proxy/core` as the value for the parameter.

The **[icapserver.groups]** section contains the default group parameters:

Priority – group priority. If a request's parameters match several groups, the processing will use the rules of the group with the highest priority.

ClientIP – IP address of the client that has requested an object through the proxy server. Objects requested from a specified IP address and located at an address defined by the **URL** parameter will be processed using the rules of this group. The following information can be used to specify the value for the **ClientIP** parameter:

- IP addresses

```
ClientIP=192.168.12.1
```

- Network addresses

```
ClientIP=192.168.12.0/24
```

```
ClientIP=192.168.12.1/255.255.255.0
```

Regular expressions are also allowed as the values, e.g.:

```
ClientIP=^192\.168\.12\..*
```

- the value will represent all addresses within the 192.168.12.0 – 192.168.12.255 range.

To define different IP addresses, you can specify the **ClientIP** parameter several times, for example:

```
[icapserver.groups]
ClientIP=192.168.20.1/24
ClientIP=192.168.30.1
```

URL – URL of a requested object. Objects with a specified URL and requested from an IP address defined by the **ClientIP** parameter will be processed using the rules of this group.



When specifying a URL, you should enter the `^.[${}]|*+?{\` symbols between the `\` escape characters. Thus, for example, the correct form of the `www.example.com` address will look as follows:

```
URL=www\.example\.com
```

The notation is compliant with the POSIX Extended Regular Expression syntax.

The **[icapserver.filter]** section contains filtration parameters for the default group:

ExcludeMimeType – exception mask for filtering by MIME type (regular expressions can be used). The application will not perform anti-virus scanning of objects with a MIME type which matches the specified mask.

ExcludeURL – exception mask for filtering by URL type (regular expressions can be used). The application will not perform anti-virus scanning of objects with URLs which match the specified mask.



You can define more than one exception mask using masks with **ExcludeURL** and **ExcludeMimeType** by specifying these parameters several times in a section, for example:

ExcludeURL=www\example\com.*

ExcludeURL=www\localsite\local.*

MaxRequestLength – maximum size of the objects to be scanned.

The **[icapserver.engine.options]** section contains the anti-virus scanning parameters for the default group:

ScanPacked=yes|no – instruction to scan packed files. To disable this mode, set the parameter to **no**.

ScanArchives=yes|no – instruction to check archived objects. To disable this mode, set the parameter to **no**.

ScanMailBases=yes|no – instruction to scan email databases (requested or transferred through the proxy server). To disable this mode, set the parameter to **no**.

ScanMailPlain=yes|no – instruction to scan databases of email messages in plain text format (requested or transferred through proxy server). To disable this mode, set the parameter to **no**.

UseHeuristic=yes|no – instruction to use heuristic analyzer during anti-virus scanning. To disable this mode, set the parameter to **no**.

Cure=yes|no – instruction to cure infected objects. To disable this mode, set the parameter to **no**.

UseAVbasesSet=standard|extended – the set of the anti-virus databases used by the application. The **extended** set contains, in addition to the records of the **standard** set, the signatures of other potentially dangerous software such as adware and remote administration utilities.

MaxScanTime – maximum time to spend scanning a single object. If an object is not checked within the specified interval, it will be assigned the **OK** status.

The **[icapserv.actions]** section contains the settings specifying actions to be taken on scanned objects for the default group:

CuredAction – action on disinfected objects.

InfectedAction – action on infected objects.

SuspiciousAction – action on suspicious objects.

WarningAction – action on an object resembling a known virus.

ErrorAction – action on an object which has caused a scanning error.

ProtectedAction – action on password-protected objects.

CorruptedAction – action on damaged objects.

LicenseErrorAction – action on scanned objects if the application has failed to load the license key information.

BasesErrorAction – action performed on objects if the application fails to load the anti-virus databases.

The **[icapserv.notify]** section contains notification parameters for the default group:

NotifyTemplateDir – directory where notification templates are stored.

NotifyScript – script used by the application to notify the administrator about objects prohibited for transfer through the proxy server.

The **[updater.path]** section contains the paths of directories and files necessary for the functioning of the *keepup2date* component:

BackUpPath – path to the directory where the anti-virus databases are archived during their update. This is a mandatory parameter.

UploadPatchPath – path to the directory containing application patches.

PidFile – path to the PID file, which is used to prevent the simultaneous launch of several instances of the *keepup2date* component. If the parameter is missing, the PID file will not be created. Consequently, no checks for other running instances of the component will be performed.

AVBasesTestPath – full path to the *avbasestest* utility, which validates the anti-virus databases. The application uses it immediately after downloading pdates. If the received updates are intact, they will be copied from a

temporary folder to the storage directory. If this parameter is not specified, during an update the updater will output to the console and log file a message informing that the downloaded anti-virus databases could not be checked; the updates will then be installed without further validation.



The *avbasestest* utility starts automatically: it does not require user participation.

The **[updater.options]** section contains parameters used by the *keepup2date* component:

KeepSilent=yes|no – the mode which determines whether component messages are output to the console. When the parameter is set to **yes** the component does not output reports to the console. The default value is **no**.

PostUpdateCmd – command performed immediately after an update of the anti-virus databases. The default value forces the application to reload the updated anti-virus databases automatically. Modification of this parameter is not recommended.

UseUpdateServerUrl=yes|no – parameter which defines whether the updater will use the address defined by the **UpdateServerUrl** parameter. The default value is **no**.

UseUpdateServerUrlOnly=yes|no – parameter which defines whether the updater will use only the address defined by the **UpdateServerUrl** parameter. When set to **no**, a failed attempt to update databases using the **UpdateServerUrl** address as the source will be followed by an attempt to use another address from the list of update servers. The default value is **no**.

UpdateServerUrl=http://url/ | ftp://url/ | /local_path/ – source address for updating of the anti-virus databases.

RegionSettings – the region where the user is located. It is used to select the most convenient Kaspersky Lab's update server from which to download updates to the anti-virus databases. Default value: **Russia**. To receive a list of all regions, run the *keepup2date* utility with the **-s** command line option.

ConnectTimeout – timeout (seconds) for network operations during updates of the anti-virus databases. The default value is **30**.

UseProxy=yes|no – the mode of proxy use during connection with Kaspersky Lab's update servers. When set to **no**, the proxy server will not be used. If the parameter is set to **yes**, the component will use the proxy address defined by the **ProxyAddress** parameter. If the **ProxyAddress** parameter value is undefined, then the **http_proxy** environment variable will be used. If the environment variable is not defined, a proxy server will not be used.

ProxyAddress – address of the proxy server used for connection. This parameter is defined as **http://username:password@url:port**. The **Username** and/or **password** parameters may be missing from the proxy address. If no address is specified, its value will be taken from the **http_proxy** environment variable.

PassiveFtp=yes|no – the parameter determines the use of passive FTP mode. The default value is **no**.

The **[updater.report]** section contains settings for output of reports by the *keepup2date* component:

ReportFilename – name of the file used for logging reports about the component's activity.

ReportLevel=0|1|2|3|4|9 – level of details in the report on the component's activity (**0** – Fatal, **1** – Error, **2** – Warning, **3** – Info, **4** – Activity, **9** – Debug). Default value: **4**.

Append=yes|no – instruction to append a report to the end of an existing report file. When the parameter is set to **no**, the previous file will be deleted when a new log is started. The default value is **yes**.

A.2. Macros

Kaspersky Anti-Virus allows the use of special macros, both in the template-based placeholder files which are sent to users instead of blocked objects (see section 5.3 on p. 40), and in the text of notification scripts (the **NotifyScript** parameter). Table 5 contains a description of these macros.

Table 5. Macros

Macro syntax	Description
%VIRUS_LIST%	List of viruses that an object is infected with.
%WARN_LIST%	List of objects containing code that resembles a known virus.
%SUSP_LIST%	List of objects suspected of infection with an unknown virus.
%CURED_LIST%	List of removed viruses.
%CLIENT_ADDR%	IP address of the client computer that has

	requested an object.
%URL%	Requested object URL
%ACTION%	Action performed on an object.
%VERDICT%	Object status.
%PRODUCT%	Product description.
%DATE%	Time of message creation.

A.3. *kavicapserver* return codes

0	No errors detected at component start.
30	Fatal system error.
65	Error loading the configuration file (file not found).
66	Error in the configuration file or command line parameters.
70	The component executable file is corrupted.

A.4. Command line options for *licensemanager*

Help options	
-h	Display on the console reference information about the component's command line options, and exit.
Command line options for managing license keys	
-s	Display on the console information about all installed license keys.

-c (C) <path_to_file>	Use the alternative configuration file <path_to_file>.
-k <key_file_name>	Display on the console information about the license key.
-i	Display on the console the detailed information about license parameter.
-a <path_to_file>	Install a license key.
-d <ajr>	Delete the current/additional key.

A.5. Licensemanager return codes

The *licensemanager* component may return any of the following codes while running:

0	The component has successfully completed its operation.
30	Fatal system error.
64	Licensing error.
65	Error reading the configuration file.
66	Error in command line options.
70	The component executable file is corrupted.

A.6. Keepup2date command line options

Help options	
-h	Display on the console reference information about the component's command line options, and exit.

-v	Display the application version on the console and exit.
-s	Display a list of update servers with information about their respective regions.
Update options	
-c <path_to_file>	Use the alternative configuration file <path_to_file> .
-u <directory>	Copy the application update to the local <directory> . Within the specified directory, the utility will reproduce the internal structure of an update server, enabling local computers to update from that directory.
-x <directory>	Copy updates for all products of Kaspersky Lab to the local <directory> . Within the specified directory, the utility will reproduce the internal structure of an update server, enabling local computers to update from that directory.
-b <path>	When updating, create in the <path> directory a backup copy of the anti-virus databases being updated.
-t <path>	Use the <path> directory to store temporary files.
-r	Cancel the last update. Updated databases will be replaced by their previous versions.
-k	Disable execution of the command defined by the PostUpdateCmd parameter.
-d <path_to_file>	Use the specified PID file.
-g <url>	Use the server with the specified URL as the source of updates.
Report generation options	
-l <path_to_file>	Log work results in file <path_to_file> .
-q	Disable output of messages about the utility's operation.

-e	Output fatal error messages only.
----	-----------------------------------

A.7. *Keepup2date* return codes

The *keepup2date* component may return any of the following codes while running:

0	The anti-virus databases do not need an update.
1	The anti-virus databases were updated successfully.
10	A fatal error occurred; updating was interrupted.
12	An error while rolling back to the previous version of the anti-virus databases. Rollback has been interrupted.
30	The PostUpdaterCmd command could not be executed after the databases were updated.
60	License information is missing, or no license key was found, using the path specified in the configuration file.
75	The configuration file cannot be loaded or contains errors.
128 + signal code	The application has exited upon a signal with the corresponding code.

APPENDIX B. KASPERSKY LAB

Founded in 1997, Kaspersky Lab has become a recognized leader in information security technologies. It produces a wide range of data security software and delivers high-performance, comprehensive solutions to protect computers and networks against all types of malicious programs, unsolicited and unwanted e-mail messages, and hacker attacks.

Kaspersky Lab is an international company. Headquartered in the Russian Federation, the company has representative offices in the United Kingdom, France, Germany, Japan, USA (CA), the Benelux countries, China, Poland, and Romania. A new company department, the European Anti-Virus Research Centre, has recently been established in France. Kaspersky Lab's partner network incorporates more than 500 companies worldwide.

Today, Kaspersky Lab employs more than 450 specialists, each of whom is proficient in anti-virus technologies, with 10 of them holding M.B.A. degrees, 16 holding Ph.Ds, and senior experts holding membership in the Computer Anti-Virus Researchers Organization (CARO).

Kaspersky Lab offers best-of-breed security solutions, based on its unique experience and knowledge, gained in over 14 years of fighting computer viruses. A thorough analysis of computer virus activities enables the company to deliver comprehensive protection from current and future threats. Resistance to future attacks is the basic policy implemented in all Kaspersky Lab's products. The company's products consistently remain at least one step ahead of many other vendors in delivering extensive anti-virus coverage for home users and corporate customers alike.

Years of hard work have made the company one of the top security software manufacturers. Kaspersky Lab was one of the first businesses of its kind to develop the highest standards for anti-virus defense. The company's flagship product, Kaspersky Anti-Virus, provides full-scale protection for all tiers of a network, including workstations, file servers, mail systems, firewalls, Internet gateways, and hand-held computers. Its convenient and easy-to-use management tools ensure advanced automation for rapid virus protection across an enterprise. Many well-known manufacturers use the Kaspersky Anti-Virus kernel, including Nokia ICG (USA), F-Secure (Finland), Aladdin (Israel), Sybari (USA), G Data (Germany), Deerfield (USA), Alt-N (USA), Microworld (India) and BorderWare (Canada).

Kaspersky Lab's customers benefit from a wide range of additional services that ensure both stable operation of the company's products, and compliance with specific business requirements. Our databases are updated every hour. The company provides its customers with a 24-hour technical support service, which is available in several languages to accommodate its international clientele.

B.1. Other Kaspersky Lab Products

Kaspersky Lab News Agent

The News Agent is intended for timely delivery of news published by Kaspersky Lab, notifications about the current status of virus activity, and fresh news. The program reads the list of available news feeds and their content from the Kaspersky Lab news server at specified intervals.

News Agent enables users to;

- See the current virus forecast .in the system tray
- Subscribe to and unsubscribe from news feeds
- Retrieve news from each selected feed at the specified interval and receive notifications about fresh news
- Review news on the selected feeds
- Review the list of feeds and their status
- Open full article text in your browser

News Agent is a stand-alone Microsoft Windows application that can be used independently or may be bundled with various integrated solutions offered by Kaspersky Lab Ltd.

Kaspersky® OnLine Scanner

This program is a free service provided to the visitors of Kaspersky Lab's corporate website. The service delivers an efficient online anti-virus scan of your computer. Kaspersky OnLine Scanner runs directly from your browser. This way, users receive quick responses to questions regarding potential infections on their computers. Using the service, visitors can:

- Exclude archives and e-mail databases from scanning
- Select standard/extended databases for scanning
- Save a report on the scanning results in .txt or .html formats

Kaspersky® OnLine Scanner Pro

The program is a subscription service available to the visitors of Kaspersky Lab's corporate website. The service delivers an efficient online anti-virus scan of your computer and disinfects dangerous files. Kaspersky OnLine Scanner Pro runs directly from your browser. Using the service, visitors can:

- Exclude archives and e-mail databases from scanning
- Select standard/extended databases for scanning
- Save a report on the scanning results in .txt or .html formats

Kaspersky Anti-Virus® 7.0

Kaspersky Anti-Virus 7.0 is designed to safeguard personal computers against malicious software as an optimal combination of conventional methods of anti-virus protection and new proactive technologies.

The program provides for complex anti-virus checks, including:

- Anti-virus scanning of e-mail traffic on the level of data transmission protocol (POP3, IMAP and NNTP for incoming mail and SMTP for outgoing messages), regardless of the mail client being used, as well as disinfection of e-mail databases.
- Real-time anti-virus scanning of Internet traffic transferred via HTTP.
- Anti-virus scanning of individual files, folders, or drives. In addition, a preset scan task can be used to initiate anti-virus analysis exclusively for critical areas of the operating system and start-up objects of Microsoft Windows.

Proactive protection offers the following features:

- **Controls modifications within the file system.** The program allows users to create a list of applications, which it will control on a per component basis. It helps protect application integrity against the influence of malicious software.
- **Monitors processes in random-access memory.** Kaspersky Anti-Virus 7.0 in a timely manner notifies users whenever it detects dangerous, suspicious or hidden processes or in case when unauthorized changes in active processes occur.
- **Monitors changes in OS registry** due to internal system registry control.
- **Hidden Processes Monitor** helps protect from malicious code concealed in the operating system using rootkit technologies.
- **Heuristic Analyzer.** When scanning a program, the analyzer emulates its execution and logs all suspicious activity, such as, opening or writing to a file, interrupt vector intercepts, etc. A decision is made based on this procedure regarding possible infection of the program with a virus. Emulation occurs in an isolated virtual environment which reliably protects the computer of infection.
- **Performs system restore** after malware attacks by logging all changes to the registry and computer file system and rolls them back at user's discretion.

Kaspersky® Internet Security 7.0

Kaspersky Internet Security 7.0 is an integrated solution for protection of personal computers against the major information- threats (viruses, hackers,

spam and spyware). A single interface enables users to configure and manage all the program's components.

The anti-virus protection features include:

- **Anti-virus scanning of e-mail traffic** on the level of data transmission protocol (POP3, IMAP and NNTP for incoming mail and SMTP for outgoing messages), regardless of the mail client being used. The program includes plug-ins for popular e-mail clients (such as Microsoft Office Outlook, Microsoft Outlook Express/Windows Mail, and The Bat!) and supports disinfection of their e-mail databases.
- **Real-time anti-virus scanning of Internet traffic** transferred via HTTP.
- **File system protection:** anti-virus scanning of individual files, folders or drives. In addition, the application can perform anti-virus analysis exclusively for critical areas of the operating system and Microsoft Windows start-up objects.
- **Proactive protection:** the program constantly monitors application activity and processes running in random-access memory, preventing dangerous changes to the file system and registry, and restores the system after malicious influence.

Protection against Internet-fraud is ensured by recognition of phishing attacks, thereby preventing confidential data leaks (above all passwords, bank account and credit card numbers) and blocking execution of dangerous scripts on web pages, pop-up windows and advertisement banners. The **autodialer blocking** feature helps identify software that attempts to use your modem for hidden unauthorized connections to paid phone services and blocks such activity. *Privacy Control* module keeps your confidential information secure from unauthorized access and transmission. *Parental Control* is a Kaspersky Internet Security component that monitors user access to the Internet.

Kaspersky Internet Security 7.0 **registers attempts to scan the ports of your computer**, which frequently precede network attacks, and successfully defends against typical network attacks. The program uses **defined rules as a basis** for control over all network transactions tracking all **incoming and outgoing data packets**. **Stealth Mode** (owing to the SmartStealth™ technology) **prevents computer detection from outside**. When you switch to Stealth Mode, the system blocks all network activity except for a few transactions allowed in user-defined rules.

The program employs an all-inclusive approach to anti-spam filtering of incoming e-mail messages:

- Verification against black and white lists of recipients (including addresses of phishing sites)
- Inspection of phrases in message body
- Analysis of message text using a learning algorithm

- Recognition of spam sent in image files

Kaspersky Anti-Virus Mobile

Kaspersky® Anti-Virus Mobile provides antivirus protection for mobile devices running Symbian OS and Microsoft Windows Mobile. The program provides comprehensive virus scanning, including:

- **On-demand scans** of the mobile device's onboard memory, memory cards, an individual folder, or a specific file; if an infected file is detected, it is moved to Quarantine or deleted
- **Real-time scanning** – all incoming and outgoing files are automatically scanned, as well as files when attempts are made to access them
- **Protection from text message spam**

Kaspersky Anti-Virus for File Servers

This software package provides reliable protection for file systems on servers running Microsoft Windows, Novell NetWare, Linux and Samba from all types of malware. The suite includes the following Kaspersky Lab applications:

- [Kaspersky Administration Kit](#).
- [Kaspersky Anti-Virus for Windows Server](#).
- [Kaspersky Anti-Virus for Linux File Server](#).
- [Kaspersky Anti-Virus for Novell Netware](#).
- [Kaspersky Anti-Virus for Samba Server](#).

Features and functionality:

- *Protects server file systems in real time*: All server files are scanned when opened or saved on the server
- *Prevents virus outbreaks*;
- *On-demand scans* of the entire file system or individual files and folders;
- *Use of optimization technologies* when scanning objects in the server file system;
- *System rollback after virus attacks*;
- *Scalability of the software package* within the scope of system resources available;
- *Monitoring of the system load balance*;

- *Creating a list of trusted processes* whose activity on the server is not subject to control by the software package;
- *Remote administration* of the software package, including centralized installation, configuration, and administration;
- *Saving backup copies of infected and deleted objects* in case you need to restore them;
- *Quarantining suspicious objects*;
- *Send notifications on events* in program operation to the system administrator;
- *Log detailed reports*;
- *Automatically update* program databases.

Kaspersky Open Space Security

Kaspersky Open Space Security is a software package with a new approach to security for today's corporate networks of any size, providing centralized protection information systems and support for remote offices and mobile users.

The suite includes four programs:

- Kaspersky Work Space Security
- Kaspersky Business Space Security
- Kaspersky Enterprise Space Security
- Kaspersky Total Space Security

Specifics on each program are given below.

Kaspersky WorkSpace Security is a program for centralized protection of workstations inside and outside of corporate networks from all of today's Internet threats (viruses, spyware, hacker attacks, and spam).

Features and functionality:

- *Comprehensive protection from viruses, spyware, hacker attacks, and spam*;
- *Proactive Defense* from new malicious programs whose signatures are not yet added to the database;
- *Personal Firewall* with intrusion detection system and network attack warnings;
- *Rollback for malicious system modifications*;
- *Protection from phishing attacks and junk mail*;

- *Dynamic resource redistribution* during complete system scans;
- *Remote administration* of the software package, including centralized installation, configuration, and administration;
- *Support for Cisco[®] NAC* (Network Admission Control);
- *Scanning of e-mail and Internet traffic* in real time;
- *Blocking of popup windows and banner ads* when on the Internet;
- *Secure operation in any type of network*, including Wi-Fi;
- *Rescue disk creation tools* that enable you to restore your system after a virus outbreak;
- *An extensive reporting system* on protection status;
- *Automatic database updates*;
- *Full support for 64-bit operating systems*;
- *Optimization of program performance on laptops* (Intel[®] Centrino[®] Duo technology);
- *Remote disinfection capability* (Intel[®] Active Management, Intel[®] vPro[™]).

Kaspersky Business Space Security provides optimal protection of your company's information resources from today's Internet threats. Kaspersky Business Space Security protects workstations and file servers from all types of viruses, Trojans, and worms, prevents virus outbreaks, and secures information while providing instant access to network resources for users.

Features and functionality:

- *Remote administration* of the software package, including centralized installation, configuration, and administration;
- *Support for Cisco[®] NAC* (Network Admission Control);
- *Protection of workstations and file servers from all types of Internet threats*;
- *iSwift technology to avoid rescanning files within the network*;
- *Distribution of load among server processors*;
- *Quarantining suspicious objects* from workstations;
- *Rollback for malicious system modifications*;

- *scalability of the software package within the scope of system resources available;*
- *Proactive Defense* for workstations from new malicious programs whose signatures are not yet added to the database;
- *Scanning of e-mail and Internet traffic* in real time;
- *Personal Firewall* with intrusion detection system and network attack warnings;
- *Protection while using Wi-Fi networks;*
- *Self-Defense from malicious programs;*
- *Quarantining* suspicious objects;
- *automatic database updates.*

Kaspersky Enterprise Space Security

This program includes components for protecting linked workstations and servers from all today's Internet threats. It deletes viruses from e-mail, keeping information safe while providing secure access to network resources for users.

Features and functionality:

- *Protection of workstations and file servers from viruses, Trojans, and worms;*
- *Protection of Sendmail, Qmail, Postfix and Exim mail servers;*
- *Scanning of all e-mails on Microsoft Exchange Server, including shared folders;*
- *Processing of e-mails, databases, and other objects for Lotus Domino servers;*
- *Protection from phishing attacks and junk mail;*
- *preventing mass mailings and virus outbreaks;*
- scalability of the software package within the scope of system resources available ;
- Remote administration of the software package, including centralized installation, configuration, and administration;
- Support for Cisco ® NAC (Network Admission Control);
- *Proactive Defense* for workstations from new malicious programs whose signatures are not yet added to the database ;

- Personal Firewall with intrusion detection system and network attack warnings ;
- *Secure operation while using Wi-Fi networks;*
- *Scans Internet traffic* in real time;
- *Rollback for malicious system modifications;*
- *Dynamic resource redistribution* during complete system scans;
- Quarantining suspicious objects ;
- *An extensive reporting system* on protection system status;
- *automatic database updates.*

Kaspersky Total Space Security

This solution monitors all inbound and outbound data streams (e-mail, Internet, and all network interactions). It includes components for protecting workstations and mobile devices, keeps information safe while providing secure access for users to the company's information resources and the Internet, and ensures secure e-mail communications.

Features and functionality:

- *Comprehensive protection from viruses, spyware, hacker attacks, and spam* on all levels of the corporate network, from workstations to Internet gateways;
- Proactive Defense for workstations from new malicious programs whose signatures are not yet added to the database ;
- *Protection of mail servers and linked servers;*
- *Scans Internet traffic* (HTTP/FTP) entering the local area network in real time;
- scalability of the software package within the scope of system resources available ;
- *Blocking access from infected workstations;*
- *Prevents virus outbreaks;*
- *Centralized reporting on protection status;*
- Remote administration of the software package, including centralized installation, configuration, and administration;
- *Support for Cisco® NAC* (Network Admission Control);
- *Support for hardware proxy servers;*

- *Filters Internet traffic* using a trusted server list, object types, and user groups;
- *iSwift technology to avoid rescanning files within the network* ;
- *Dynamic resource redistribution during complete system scans*;
- *Personal Firewall with intrusion detection system and network attack warnings* ;
- *Secure operation for users on any type of network*, including Wi-Fi;
- *Protection from phishing attacks and junk mail*;
- *Remote disinfection capability* (Intel[®] Active Management, Intel[®] vPro™);
- *Rollback for malicious system modifications*;
- *Self-Defense from malicious programs*;
- *full support for 64-bit operating systems*;
- *automatic database updates*.

Kaspersky Security for Mail Servers

This program is for protecting mail servers and linked servers from malicious programs and spam. The program includes application for protecting all standard mail servers (Microsoft Exchange, Lotus Notes/Domino, Sendmail, Qmail, Postfix and Exim) and also enables you to configure a dedicated e-mail gateway. The solution includes:

- [Kaspersky Administration Kit](#).
- [Kaspersky Mail Gateway](#).
- [Kaspersky Anti-Virus for Lotus Notes/Domino](#).
- [Kaspersky Anti-Virus for Microsoft Exchange](#).
- [Kaspersky Anti-Virus for Linux Mail Server](#).

Its features include:

- *Reliable protection from malicious or potentially dangerous programs*;
- *Junk mail filtering*;
- *Scans incoming and outgoing e-mails and attachments*;

- *Scans all e-mails on Microsoft Exchange Server for viruses, including shared folders;*
- *Processes e-mails, databases, and other objects for Lotus Notes/Domino servers;*
- *Filters e-mails by attachment type;*
- *Quarantines suspicious objects;*
- *Easy-to-use administration system for the program;*
- *Prevents virus outbreaks;*
- *Monitors protection system status using notifications;*
- *Reporting system for program operation;*
- *scalability of the software package within the scope of system resources available ;*
- *automatic database updates.*

Kaspersky Security for Internet Gateways

This program provides secure access to the Internet for all an organization's employees, automatically deleting malware and riskware from the data incoming on HTTP/FTP. The solution includes:

- [Kaspersky Administration Kit](#).
- [Kaspersky Anti-Virus for Proxy Server](#).
- [Kaspersky Anti-Virus for Microsoft ISA Server](#).
- [Kaspersky Anti-Virus for Check Point FireWall-1](#).

Its features include:

- *Reliable protection from malicious or potentially dangerous programs;*
- *Scans Internet traffic (HTTP/FTP) in real time;*
- *Filters Internet traffic using a trusted server list, object types, and user groups;*
- *Quarantines suspicious objects;*
- *Easy-to-use administration system;*
- *Reporting system for program operation;*
- *Support for hardware proxy servers;*

- Scalability of the software package within the scope of system resources available ;
- *Automatic database updates.*

Kaspersky® Anti-Spam

Kaspersky® Anti-Spam is a cutting-edge software suite designed to help organizations with small- and medium-sized networks wage war against the onslaught of unsolicited e-mail messages (spam). The product combines the revolutionary technology of linguistic analysis with modern methods of e-mail filtration, including DNS Black Lists and formal letter features. Its unique combination of services allows users to identify and wipe out up to 95% of unwanted traffic.

Installed at the entrance to a network, where it monitors incoming e-mail traffic streams for spam, Kaspersky® Anti-Spam acts as a barrier to unsolicited e-mail. The product is compatible with any mail system and can be installed on either an existing mail server or a dedicated one.

Kaspersky® Anti-Spam's high performance is ensured by daily updates to the content filtration database, adding samples provided by the Company's linguistic laboratory specialists. Databases are updated every 20 minutes.

Kaspersky Anti-Virus® for MIMESweeper

Kaspersky Anti-Virus® for MIMESweeper provides high-speed scanning of traffic on servers running Clearswift MIMESweeper for SMTP / Clearswift MIMESweeper for Exchange / Clearswift MIMESweeper for Web.

The program is a plug-in and scans for viruses and processes inbound and outbound e-mail traffic in real time.

B.2. Contact Us

If you have any questions, comments, or suggestions, please refer them to one of our distributors or directly to Kaspersky Lab. We will be glad to assist you in any matters related to our product by phone or via e-mail. Rest assured that all of your recommendations and suggestions will be thoroughly reviewed and considered.

Technical support	Please find the technical support information at http://www.kaspersky.com/supportinter.html Helpdesk: www.kaspersky.com/helpdesk.html
General information	WWW: http://www.kaspersky.com http://www.viruslist.com E-mail: info@kaspersky.com