KASPERSKY LAB

Kaspersky Anti-Virus® 5.5 for Linux / FreeBSD / OpenBSD Workstations and File Servers

ADMINISTRATOR'S GUIDE

KASPERSKY ANTI-VIRUS® 5.5 FOR
LINUX / FREEBSD / OPENBSD WORKSTATIONS AND FILE
SERVERS

Administrator's guide

http://www.kaspersky.com

Revision date: May 2005

Table of contents

Chapter 1. Kaspersky Anti-Virus 5.5® for Linux/FreeBSD/OpenBSD Workstations and File Servers

Kaspersky Anti-Virus for Linux/FreeBSD/OpenBSD Workstations and File Servers (hereinafter also referred to as Kaspersky Anti-Virus, or the application) protects file servers and workstations running Linux, FreeBSD, or OpenBSD operating systems from viruses and malware.

The application performs these key functions:

- *Real-time protection of files against malicious code:* calls to the file system are intercepted and analyzed by the anti-virus core; infected objects are disinfected or removed, and suspicious objects are isolated for further analysis.
- *On-demand scanning of files*: search for infected or suspicious objects (including an option to define scanning areas); their analysis; disinfection, removal or isolation of objects for further examination.
- *Quarantine of suspicious and corrupted objects:* suspicious objects are moved to a quarantine directory.
- *Creation of backup copies for infected objects prior to disinfection or removal* in order to allow on-demand restoration of objects containing valuable data.
- *Updating of the anti-virus databases and application modules* from Kaspersky Lab's update servers.
- *Administration and remote setup of Kaspersky Anti-Virus* via the web interface provided by the Webmin program and the application configuration file.

1.1. What's new in version 5.5

Version 5.5 of **Kaspersky Anti-Virus for Linux/FreeBSD/OpenBSD Workstations and File Servers** has the following improvements over version 5.0:

- The application package now includes a new *kavmonitor* component ensuring real-time anti-virus protection for files.
- New technologies have been implemented for downloading of updates to the anti-virus databases and application modules (including integrity checks for downloaded databases and their further distribution) thus ensuring considerable network traffic economy.
- The active anti-virus database (standard databases, extended or paranoid set) can be specified individually for each application component.
- The application setup and removal procedures have been considerably simplified.
- Installation has been made faster by allowing the application to import configuration settings from earlier versions (4.0 or 5.0).
- A backup storage area preserves copies of suspicious or infected objects prior to their disinfection or removal. This allows the recovery of valuable data if errors occur during object disinfection.
- The iChecker database technology and double-level caching of scanned objects have been implemented to decrease CPU load during anti-virus scanning.
- The option to restrict the number of objects scanned simultaneously in the background has been added, to optimize computer load.
- A list of detectable viruses can now be generated.

1.2. Licensing policy

Kaspersky Anti-Virus licensing policy includes a system limiting application use based on duration, usually to one year from the date of purchase.

1.3. Hardware and software requirements

The minimum system requirements for **Kaspersky Anti-Virus are**:

- Hardware requirements:
  - Intel Pentium-class processor.
  - 32 Mb of free RAM or more.
  - 100 Mb or more of available hard disk space.
- Software requirements:
  - One of the following operating systems:
    - RedHat Linux versions 9.0, Fedora Core 2, Advanced Server 3
    - SuSE Linux versions Enterprise Server 9.0, 9.2
    - Debian GNU/Linux version 3.0 updated (r4)
    - Mandrake Linux version 10.1
    - FreeBSD versions 4.10 or 5.2.1
    - OpenBSD version 3.6
  - The Webmin program (www.webmin.com) for remote administration of Kaspersky Anti-Virus.
  - Perl version 5.0 or higher, for Kaspersky Anti-Virus installation using *install.sh*.
  - Installed compiler packages (gcc, binutils, glibc-devel, make, ld) and installed operating system source code, required to use the *kavmonitor* component.

1.4. Distribution kit

You can purchase the software from our distributors (retail box), or from one of our web shops (for example, www.kaspersky.com, **E-Store** section).

When purchasing a retail box you will receive the following distribution kit:

- A sealed envelope with an installation CD containing software application files

- Administrator's guide
- License key file included in the program distribution package on the CD or stored on a special floppy disk
- License agreement.

Please read the license agreement carefully before opening the CD envelope.
Opening the sealed envelope of the installation CD or installing the application on a computer confirms your acceptance of all terms and conditions of the license agreement.

If you purchase our application from a web shop, you will download it from Kaspersky Lab's website; the copy also contains this manual. Your license key is either included in the installation package or will be sent to you by e-mail after payment.

The license agreement constitutes a legal agreement between you and Kaspersky Lab containing the terms and conditions under 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 must return the box containing Kaspersky Anti-Virus to the distributor where you purchased it; you will be refunded the amount you've paid for subscription, provided the CD envelope remains sealed.

1.5. Services for registered users

Kaspersky Lab offers its legal users a broad range of services maximizing the efficiency of Kaspersky Anti-Virus software.

By purchasing a subscription, you become a registered software user entitled to the following services throughout the license period:

- software upgrades for this software application;
- consultations regarding issues pertaining to installation, configuration and use of this software, available over the telephone or via e-mail;
- 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 e-mail newsletter service.

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

1.6. Adopted conventions

The text in this document uses various styles depending upon its purpose. The table below lists adopted conventions used in the text.

Style	Purpose
Bold type	Used to indicate menu titles, menu items, window titles, parts of dialog boxes, and other graphical interface items.
Italic type	Used to indicate program components.
Note.	Additional information, notes.
Attention!	Information requiring special attention.
In order to perform the action, 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 keys.
`Text of information messages and the command line`	Text of configuration files, information messages and the command line.

Chapter 2. Installing Kaspersky Anti-Virus

Before installing Kaspersky Anti-Virus, you are advised to make the following preparations for your system:

- Make sure that the system meets the minimum hardware and software requirements listed in section 1.3 on p. 8. If any application, e.g., Perl, has not been installed yet, you are advised to install it, or else a part of the application's functionality will be unavailable.
- Configure your Internet connection.
- Log in to the system as **root**.

2.1. Installing the application on a computer running Linux

Kaspersky Antivirus is distributed in three different installation packages for Linux systems; which you use depends upon the type of your Linux distribution.

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

```
rpm -i <distribution_package_filename>
```

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

```
dpkg -i <distribution_package_filename>
```

You can also use the universal distribution file intended for all Linux operating systems. Use this file if your Linux version does not support the RPM or DEB formats, or if your network administrator does not use a built-in package manager.

The universal installation package of Kaspersky Anti-Virus is supplied as an archive. The archive contains the directory tree with the distribution package files and the *install.sh* installation script which carries out the installation.

To install Kaspersky Anti-Virus from the universal installation package, please do the following:

1. Copy the archived installation package to a directory in the computer file system and unpack it.

2. Run the nstallation script: *./install.sh* .

2.2. Installing the application on a computer running FreeBSD or OpenBSD

The installation package of Kaspersky Anti-Virus is supplied in a .pkg package for computers running FreeBSD or OpenBSD operating systems.

To initiate installation of Kaspersky Anti-Virus from the .pkg package, enter the following on the command line:

```
pkg_add <package_name>
```

2.3. Overview of installation procedure

To install the application, follow the steps below:

1. Copy the application files to the computer.

2. Install a license key.

   If the license key is not installed, the configuration process will not start and work with the application will be impossible. If you have no license key at the time of installation (for example, if you purchased the application via the Internet and have not received the license key by e-mail yet), you can install the key after the setup procedure but before you actually start using the application.

3. Configure the *keepup2date* component.

4. Install and update the anti-virus databases.

Ensure that the anti-virus databases are installed before you begin using the application. The procedures of anti-virus scanning and disinfection depend upon the records in the anti-virus databases containing descriptions of all currently known viruses and cure methods for infected objects. Scanning and processing of files cannot be performed without the anti-virus databases!

5. Install the Webmin module.

   The Webmin module for remote software management can be installed correctly only if the Webmin application is located in the default directory. After Webmin is installed, you will receive detailed instructions on how to configure the module to work with the application.

After software installation the appropriate kernel module has to be compiled and installed to enable real-time scanning!

2.4. Updating a previous version

After the application is installed, the system is searched for earlier versions of Kaspersky Anti-Virus.

If the installer detects application version earlier than 5.5, it imports some of its settings into the current version's configuration file.

The installer does not remove the distribution package of the previous version of Kaspersky Anti-Virus. That task must be performed by the administrator.

Some standard parameters of the configuration file (e.g. the path to the directory containing the anti-virus databases) are not imported. Instead they are determined during the setup procedure.

Furthermore, individual components of the current application version have more options than the corresponding components in versions 4.0 and 5.0. You are advised to check the correctness of the configuration file before using the application.

2.5. License key installation

During this stage of the setup process, the installer searches the current directory for the license key, i.e. the file with the *.key* extension required for the operation of Kaspersky Anti-Virus. The file enables complete functionality of the application. It is impossible to use Kaspersky Anti-Virus until the license key is installed.

If the license key is detected, the installer prints the corresponding message to the console and proceeds to the next stage, which is installation of the anti-virus databases (see section 3.2 on p. 16).

If the license key is not detected, the installer requests that you specify its full path. If no key is available (for example, the application was purchased on-line and the license key has not been received yet) skip the step of specifying the license key path by choosing **[cancel]**, and proceed with the application installation.

When you receive the license key, it must be copied to the key storage directory specified in the **LicensePath** parameter of Kaspersky Anti-Virus configuration file.

2.6. Completing the installation

If all the installation steps described above are finished successfully, a message confirming that success will be printed to the console. The configuration file included in the distribution package contains all the settings required to start the program. A number of file parameters are determined during program installation while others are specified by default (see section 3.1 on p. 15). However, the administrator will have to adjust some settings to begin working with Kaspersky Anti-Virus. For more information about the settings required before you can use the application, please refer to Chapter 3 on p. 15.

If any installation step has been skipped (e.g., there was no access to the anti-virus databases during the setup), you can carry it out later.

Chapter 3. Post-install setup

The installation routine performs analysis of the system on which Kaspersky Anti-Virus is being installed, and automatically determines some configuration parameters by setting them to sensible default values (please see section 3.1 on p. 15).

You are advised to start using the application by configuring it to work with the Webmin package (please see details on Webmin tuning, in that product's documentation).

In this chapter we shall examine the default settings of Kaspersky Anti-Virus and review in detail the configuration file settings.

3.1. Default application settings

This section reviews the application's default parameters, which are defined in the default configuration file *kav4unix.conf*. The information in this section will enable you to determine whether Kaspersky Anti-Virus needs additional tuning (please see Chapter 5 on p. 34) for its more efficient integration into your corporate environment.

You can create your own configuration files, and use them both for particular tasks and as the default configuration.

By default Kaspersky Anti-Virus is configured to launch the real-time protection component *kavmonitor* when the operating system boots. If the on-demand scanning component *kavscanner* is started without additional command line keys, it starts scanning computer directories and file systems for viruses beginning with the current one.

If infected, suspicious or corrupted files are discovered, corresponding notifications are output to console and appended to the report file.

Please note that discovered infected objects ARE NOT CURED BY DEFAULT!

3.2. Installing/updating the anti-virus databases

The application detects viruses and cures infected objects using the records in its anti-virus databases, which contain descriptions of all currently known viruses and methods to be used for their disinfection.

It is essential to keep the anti-virus databases up-to-date, because new viruses emerge daily. You are advised to update the anti-virus databases **immediately** when the application is installed because the databases in the distribution package may be obsolete by the time of installation.

To update the databases, launch the *keepup2date* component. Type the following on the command line:

```
/path/to/ keepup2date
```

The anti-virus databases will be downloaded from Kaspersky Lab's update servers to a special directory specified in the configuration file.

The anti-virus databases of Kaspersky Lab are updated every hour. You are advised to update the anti-virus databases at least EVERY THREE HOURS to keep the application up-to-date and switch to HOURLY updates during virus outbreaks.

Furthermore, if you have upgraded Kaspersky Anti-Virus from an earlier version, you are advised to take additional action (please see section 4.1.1 on p. 19 for details). This is necessary because the component previously responsible for updating the anti-virus databases has been replaced.

3.3. Setting the application up for work with Webmin

If you plan to configure Kaspersky Anti-Virus remotely you are advised to configure it up to work with the Webmin package.

For example, Webmin may be used for restricting access to the program through a system of user passwords.

All application settings modified remotely from Webmin are saved to the default configuration file of the application.

If you wish to create an alternative configuration file using Webmin, you'll have to perform the following actions:

1. Copy the data from the existing configuration file to a new one saving it under a different name. Then modify the new (alternative) configuration file as required.
2. Specify the alternative configuration file name in the **Full path to KAV config** parameter entry field of the **Config edit** tab.

Please refer to the **Webmin** documentation for details on its settings. You should also use Webmin help for answers to any questions about the application's **remote administration module**.

Hereafter, while discussing the application settings and task launch, we **shall not** specifically describe the procedure for remote operations via Webmin**.**

Chapter 4. Using Kaspersky Anti-Virus

Kaspersky Anti-Virus creates a system of anti-virus protection for your computer, which covers the whole range of objects from individual files to the entire file system.

The tasks which administrators can perform using Kaspersky Anti-Virus can be subdivided into the following groups:

1. Updating the anti-virus databases used for scanning and cleaning of infected objects.
2. Anti-virus protection of file systems on a computer, via scheduled and/or on-demand scanning.
3. Constant anti-virus protection i.e. real-time protection.

Each of these groups consists of more specific tasks implementing particular functions of the application. In this chapter, we shall discuss the most interesting tasks, which the administrator can combine or elaborate upon for a particular business setting.

⚠ In all the task descriptions further we shall assume that the administrator has performed due post-install application setup (please see Chapter 3 on p. 15).

4.1. Updating the anti-virus databases

The application's *keepup2date* component performs the essential function of maintaining the current status of the anti-virus databases, which are used by Kaspersky Anti-Virus while scanning for, and cleaning, infected files. They can be downloaded from Kaspersky Lab's update servers, at these addresses:

http://downloads1.kaspersky-labs.com/updates/
http://downloads2.kaspersky-labs.com/updates/
ftp://downloads1.kaspersky-labs.com/updates/ and other servers.

A full list of addresses from which updates can be downloaded can be found in the *updcfg.xml* file included in the application package. The list will be updated automatically on a regular basis.

During the updating procedure, the *keepup2date* component accesses the server list in this file, selects an address and attempts to download the anti-virus databases from the corresponding server. If the attempt to update fails, *keepup2date* repeats the process using the next address. After a successful update the application restarts automatically by default (**PostUpdateCmd** parameter in the **[updater.options]** section).

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

If the structure of your LAN is rather complicated, you are advised to download the anti-virus database updates to a network directory and configure other networked computers to copy the updates from that directory.

We strongly recommend that you update the anti-virus databases at least every 3 hours!

The updating procedure can be scheduled using the **cron** utility, or the administrator may choose to run it manually (from the command line).

4.1.1. *keepup2date* application component

The *keepup2date* component, which updates the anti-virus databases in version *5.5* of Kaspersky Anti-Virus, replaces the *kavupdater* component used by previous versions. The new component has improvements to *kavupdater's* functions and some new features:

- the option to automatically select the geographically closest update server, based on the region specified in the configuration file;
- the option to download and install incremental updates when cumulative updates become available, which may be useful for traffic economy;
- the ability to recover from either a disconnection while downloading the anti-virus databases, or an updates' server change. After reconnection the component only downloads the remaining portion of the anti-virus databases instead of starting from the beginning;
- an integrity check for downloaded databases;
- the option to launch a user-defined command to reload the anti-virus databases immediately after a successful update;
- the option to roll back to the previous version of the anti-virus databases;

- the *wget* program is no longer required by *keepup2date*;
- LAN computers can be updated from a shared directory, either on a Samba server, or on a computer running Microsoft Windows;
- the component allows selection of the anti-virus database set (standard set, extended or paranoid sets).

There are several parameters specific for the *kavupdater* component which are no longer necessary, and which will be removed automatically from the application's configuration file.

However, if you have upgraded Kaspersky Anti-Virus from version 4.0, you will have to remove some parameters from the configuration file *manually*.

Parameters which are no longer necessary and therefore should be removed from the application configuration file can be found in the **[updater.options]** section:

- **RandomServerOrder** – the option is not available because the server selection procedure has changed.
- **ReloadApplication** – the option has been replaced with a more general one, which allows execution of the **PostUpdateCmd** script;
- **ExtraWgetOptions** – the component no longer uses *wget* as an external program.
- **ShowExternalCmdOutput** – the new component does not execute external commands.

Additionally, the new component does not use the **UpdateServersFile** option in the **[path]** section, because the list of servers is now updated dynamically.

Please see the corresponding man pages for details on the options used by the *keepup2date* component.

The anti-virus databases can be updated in several ways. Let us examine them closely.

Task: configure the application to download updates to its anti-virus databases from Kaspersky Lab's update servers. The update server's address should be selected automatically from the list provided with the *keepup2date* component.

Solution: to accomplish the task, do the following:

Assign the **No** value to the **UseUpdateServerUrl** parameter in the **[updater.options]** section.

Task: configure the application to download updates to its anti-virus databases from the address specified by the administrator. The process should be terminated if downloading from the specified address fails.

Solution: to accomplish the task, do the following:

Assign the **Yes** values to the **UseUpdateServerUrl** and **UseUpdateServerUrlOnly** parameters in the **[updater.options]** section. Additionally, the **UpdateServerUrl** parameter should contain the update server's address.

Task: configure the application to download updates to its anti-virus databases from the address specified by the administrator. If an update from the specified address fails, the databases must be updated using another address from the updater component's internal list of servers.

Solution: to accomplish the task, do the following:

Assign the value **Yes** to the **UseUpdateServerUrl** parameter in the **[updater.options]** section, and **No** to the **UseUpdateServerUrlOnly** parameter. Additionally, the **UpdateServerUrl** parameter should contain the update server's address.

4.1.2. Scheduling anti-virus database updates using cron

Regular automatic anti-virus database updates can be scheduled using the **cron** utility.

Task: schedule automatic anti-virus database updating to run every 3 hours. The system log should be updated with operational application errors only. A general log should list every commencement of the task, with no information output to console.

Solution: to accomplish the task:

1. Set the appropriate values in the application configuration file:

```
[updater.options]
KeepSilent=yes
[updater.report]
Append=yes
ReportLevel=1
```

2. Modify the file which defines rules for the **cron** process (**crontab –e**), adding the following line:

   ```
   0 0-23/3 * * * /opt/kav/bin/keepup2date
   ```

4.1.3. Manual updating of the anti-virus databases

The procedure for updating the anti-virus databases can be started at any time from the command line.

Task: start the update procedure for the anti-virus databases and report the results in the */tmp/updatesreport.log* file.

Solution: enter the following at the command line:

```
keepup2date –l /tmp/updatesreport.log
```

If you need to update anti-virus databases on several computers it is usually more convenient for one server to download the databases and save them to a network directory, and update all computers from that directory, rather than for each computer to download the files individually.

Task: configure updating of the anti-virus databases using files in the **/home/bases** network directory**.** If the directory is inaccessible or empty, the updating procedure should use Kaspersky Lab's update servers. Output the results of the work to a report file.

Solution: to accomplish the task:

1. Set the appropriate values in the application configuration file:

   ```
   [updater.options]
   UpdateServerUrl=/home/bases
   UseUpdateServerUrl=yes
   UseUpdateServerUrlOnly=no
   ```

2. Enter the following at the command line:

   ```
   keepup2date –l /tmp/report.txt
   ```

4.1.4. Creating a network directory for the anti-virus databases

To ensure that updates to the anti-virus databases are distributed correctly from a shared network directory in your LAN to local computers, the file structure in the network directory must be identical to the structure of Kaspersky Lab's update servers. This section discusses the task of creating a network directory.

Task: create a network directory to be used as a source of updates by LAN computers.

Solution: to accomplish the task:

1. Create a local directory.
2. Launch the *keepup2date* component as follows:

   ```
   keepup2date -u <rdir>
   ```

   where `rdir` stands for a complete path to the created directory.
3. Grant computers on the LAN network access to that directory.

Task: set up updating of the anti-virus databases through a proxy server.

Solution: in order to accomplish the task, do the following:

1. Assign the value **Yes** to the **UseProxy** parameter in the **[updater.options]** section of the configuration file**.**
2. Ensure that the **ProxyAddress** parameter in the **[updater.options]** section of the configuration file contains the proxy server address. The address must be specified in the following format: **http://username:password@ip_address:port.** The **ip_address** and **port** values are obligatory, while **username** and **password** have to be specified only if the proxy requires authorization.

or:

1. Assign the value **Yes** to the **UseProxy** parameter in the **[updater.options]** section of the configuration file.

2. Specify the **http_proxy** environment variable in the following format: **http://username:password@ip_address:port**. Please note that this variable will be taken into account only if the **UseProxy** parameter in the **[updater.options]** section is either missing or set to **Yes**.

The method specified for **PassiveFtp** use, specified by the **PassiveFtp** parameter in the **[updater.options]** section of the configuration file, must correspond to the settings of your external firewall!

4.2. Anti-virus protection of file systems

Computer file systems are protected against viruses by the *kavscanner* component, which scans computer files for viruses, and processes infected and/or suspicious objects in accordance with its settings. The processing may be either purely informational (information sent to a report and console, or to the administrator), or it may modify the object (disinfection, relocation to quarantine, or removal).

All the settings of the *kavscanner* component are grouped in the **[scanner.*]** section of the application's configuration file.

On-demand scanning of your computer file systems may be invoked from the command line or scheduled using the standard **cron** utility. The range of the file system to be scanned can be specified, from the whole file system down to individual directories or objects.

The following sections discuss in detail the most typical tasks of anti-virus protection for computer file systems.

The process of scanning a whole computer for viruses is a very resource-consuming task. Please keep in mind that while it is running, the overall computer performance drops, and therefore running any other processes at the same time is not recommended. To avoid these problems, you are advised to scan individual directories instead.

4.2.1. On-demand scanning of files in an individual directory

Kaspersky Anti-Virus enables scanning for, and disinfection of, files in specified directories.

Task: start scanning the **/tmp** directory, automatically disinfecting all infected objects. Objects which cannot be disinfected, are to be deleted.

Create files *infected.lst*, *suspicion.lst*, *corrupted.lst* and *warning.lst* in the same directory, recording the file names of, respectively, infected, suspicious or corrupted objects revealed by the scanning procedure.

Store the results of component activity (start date, detailed information about all files except for those containing no viruses) in a report file *kavscanner-<<current_date>>-pid.log* in the same directory.

Solution: in order to accomplish the task you should enter the following in the command line:

```
#kavscanner -rlq -pi /tmp/infected.lst
-ps /tmp/suspicion.lst -pc /tmp/corrupted.lst
-pw /tmp/warning.lst -o /tmp/kavscanner-`date
"%Y-%m-%d-$$"`.log -i3 -ePASBMe -j3 -mCn /tmp
```

4.2.2. Scheduled directory scanning

The standard Unix scheduling utility **cron** can be used for the automated performance of any Kaspersky Anti-Virus tasks.

Task: schedule daily scanning for virus presence, to start at 0 hrs. 00 min. for the **/home** directory, using scanning parameters defined in the */etc/kav/scanhome.conf* configuration file.

Solution: in order to accomplish the task, do the following:

1. Create the */etc/kav/scanhome.conf* configuration file and record all required scanning parameters in it (please see section 5.1 on p. 34 for details).

2. Modify the file which defines the tasks for **cron**, by typing **crontab –e** at the command line, and add the following line:

   ```
   0 0 * * * /path/to/kavscanner -c
   /etc/kav/scanhome.conf /home
   ```

4.2.3. Moving objects to a separate directory (quarantine)

Kaspersky Anti-Virus can be configured to move all infected objects found within the computer's file system to a special directory.

Such an approach can be used, for example, if during the antiviral scanning of a directory an infected file containing important data is detected. Since part of the data may get lost during disinfection, an appropriate approach may be to isolate the infected object in a special directory for subsequent sending to Kaspersky Lab for analysis.

If you intend to keep the Quarantine directory within the computer's file system, we advise that you exclude it from the target area for all subsequent scans by specifying its full path in the **ExcludeDir** parameter of the **[scanner.options]** section in the configuration file.

Task: scan all the objects listed in the */tmp/download.lst* file, moving any infected objects to the */tmp/infected* directory. Record information about infected, suspicious, and corrupted objects to the report file.

Solution: in order to accomplish the task, do the following:

1. For actions to be performed on infected objects, add the following line to the **[scanner.object]** and **[scanner.container]** sections of the configuration file:

```
OnInfected=MovePath /tmp/infected
```

2. Disable the disinfection mode (**Cure=no**) if it was enabled.

3. Enter the following in the command line:

```
# kavscanner -@/tmp/download.lst -ePASBME -rq
-i0 -o /tmp/report.log -j3 -mCn
```

If it is a requirement that access to the files in the directory */tmp/infected* be limited to reading and writing, this can be accomplished using standard Unix tools (the **chmod** command) by making the following changes to the task structure:

4. Add the line below as a rule for processing of infected files in the **[scanner.object]** and **[scanner.container]** sections of the application configuration file:

```
OnInfected=exec mv %FULLPATH%/%FILENAME%
/tmp/infected/%FILENAME%; chmod -x
/tmp/infected/%FILENAME%
```

4.2.4. Additional processing: using scripts

Kaspersky Anti-Virus enables additional processing of objects, which have passed through anti-virus scanning, by using standard Unix commands and scripts. These tools allow experienced administrators to extend the functionality of Kaspersky Anti-Virus by defining different actions to be performed on objects of different status.

4.2.4.1. Disinfection of archived objects

Kaspersky Anti-Virus does not perform disinfection of compressed infected files; it just discovers suspicious and infected objects inside archives. However, this capability can be implemented using an additional script such as the *vox.sh* script, used for disinfecting *tar* and *zip* archives, which is included in the distribution package of Kaspersky Anti-Virus.

Task: scan all *tar* and *zip* archives accessible on a computer and attempt disinfection of all the objects they contain using the *vox.sh* script. Use /etc/kav/*kavscanner.conf.in* as a configuration file, where script application for disinfection of archives should be specified prior to the scanning procedure.

List all infected objects with their full paths in the */tmp/infected_archive.lst* file. Store a report of the component's activity in the */tmp/logfile.log* file.

Solution: in order to accomplish the task, do the following:

1. Create an alternative *kavscanner.conf.in* file.
2. Define the rules for processing infected objects, by adding the following line in the **[scanner.container]** section of that file:

   ```
   OnInfected=exec /tmp/kavscanner/test/vox.sh
   %FULLPATH%/%FILENAME%
   ```

3. Enter the following at the command line:

   ```
   # kavscanner -c kavscanner.conf.in -ePASE -qR
   -o /tmp/logfile.log -j3
   -pi /tmp/infected_archive.lst /
   ```

4.2.4.2. E-mail notification of administrator

Kaspersky Anti-Virus allows standard Unix tools to be configured to notify the administrator about infected, suspicious or corrupted objects discovered within computer file systems.

Task: configure notification of the administrator about infected files and archives discovered in the computer file system during each scan performed in accordance with the parameters defined in the *kav4unix.conf* configuration file.

Solution: in order to accomplish the task, do the following:

Define the rules for processing simple objects and container objects in the *kav4unix.conf* configuration file:

```
[scanner.object]
OnInfected=exec echo %FULLPATH%/%FILENAME% is
infected by %VIRUSNAME% |
mail -s kavscanner admin@localhost.ru
[scanner.container]
OnInfected=exec echo archive %FULLPATH%/%FILENAME% is
infected, viruses list is in the attached file %LIST%
| mail -s kavscanner -a %LIST% admin@localhost.ru
```

4.2.5. Backup of processed objects

If infected files are automatically deleted, as the specified action for infected files, valuable data may be lost. Data are also at risk during disinfection. To avoid this, Kaspersky Anti-Virus offers the option to copy infected files to a backup storage directory. The full path of each object is also recorded, to allow later restoration if necessary.

Prior to an object’s disinfection or removal, the application can be configured to automatically copy it to the backup storage directory, specified by the parameter **BackupPath** in the **[path]** section. Subsequent recording of the same object to backup storage automatically replaces its earlier copy with a newer one.

Please note that by default the backup mode is off and the path to the backup storage directory is not defined. The path must be specified in the configuration file to enable backup mode.

If an object is removed from the file system its backup copy will be preserved until it is deleted by the administrator.

4.3. Real-time anti-virus protection

Real-time anti-virus protection of computer file systems is accomplished by the *kavmonitor* component. Its detailed operation and different variants of the settings for real-time anti-virus protection are discussed in this section.

4.3.1. The operational algorithm of *kavmonitor component*

The following algorithm is used in the *kavmonitor* component:

1. When a program attempts to access an object within the protected file system (requesting to open, launch or close a file) the call is intercepted by the kernel module of the *kavmonitor* component, and the file is checked for the presence of a virus.

2. The intercepted file is processed using a daemon program included in the *kavmonitor* component.

3. The daemon scans the requested object, checking it for viruses and processing it in accordance with the parameters specified in the configuration file, including disinfection using the anti-virus databases if that option is enabled.

4. When the file has been processed, the daemon sends *kavmonitor* an access code (granted/denied), which defines the file's status.

5. *kavmonitor* either allows or denies access to the file, depending upon its status. In the latter case the program which has requested access to the file will receive an error code indicating that access has been denied.

As a result of the scanning, and possible disinfection, procedure a file can be assigned one of the following status values:

- **Clean** – the object is not infected.
- **Infected** – the object is infected.
- **Cured** – the object has been successfully disinfected.
- **CureFailed** – object disinfection has failed.
- **Warning** – object code resembles a known virus.

- **Suspicion** – object infection with an unknown virus is suspected.
- **Protected** – the object cannot be scanned because it is encrypted.
- **Corrupted** – the object is damaged.

Access will be blocked if the object is infected (**Infected**, **CureFailed**). In all other cases access to the file will be granted.

4.3.2. File scanning and disinfection mode

The *kavmonitor* component performs anti-virus scanning when any operations pertaining to file access (open, close or launch) are attempted. If a file is being closed, it is scanned only if it has been modified.

By default, intercepted infected files are not disinfected; the application will just automatically block access to such objects.

Disinfection of infected objects is switched on in the configuration file (section **[monitor.options]**, parameter **Cure=yes**). If after scanning a file, *kavmonitor* discovers that it is infected (i.e. file status is **Infected**), it will act in accordance with the settings in its configuration file (please see section 4.3.3 on p. 30).

Please note that files with **CureFailed** status are treated using the actions defined for infected objects.

4.3.3. Operations on suspicious and infected objects

Some actions can be defined for objects which when scanned have **Infected**, **Suspicious** or **Warning** status, such as:

- *transfer to a specified directory* – objects with a definite status can be moved to a specified directory. The two options are *regular* and *recursive* (with full path)*transfer*;
- *removal* of the file from the file system;

The rules for processing objects are set in the application configuration file's **[monitor. actions]** section.

4.3.4. Quarantine for infected objects

Infected objects are moved to a separate directory to quarantine them, allowing their subsequent restoration. They are moved when file disinfection has failed (for instance, when only two of the three viruses discovered in a file have been removed successfully). The action is controlled by the **MovePath** parameter in the **[monitor.actions]** section.

The administrator can configure the transfer of objects to different directories depending upon their status.

Task: scan all requested files for viruses and cure them if infected. If disinfection fails, transfer infected objects with their full paths to the **/tmp/infected** directory.

Solution: in order to accomplish the task, do the following:

1. Turn on the cure mode for infected objects in the application configuration file (set **Cure=yes** in the **[monitor. options]** section).
2. Define the rules for processing infected objects, by adding the following settings in the **[monitor.actions]** section of the configuration file:

```
OnInfected=MovePath /tmp/infected
```

4.4. License key management

A license key entitles you to use the application and also contains data pertaining to the license, such as license type, expiration date, information about distributors, etc.

During the period of license validity you are entitled to the following services:

- round-the-clock technical support;
- hourly updates of anti-virus databases;
- application updates (patches);
- new application versions (upgrades);
- timely notifications about new viruses.

When the license expires, these services are discontinued automatically. Kaspersky Anti-Virus will continue scanning files but it will use only the anti-virus databases which were current when the license expired, as the function of anti-virus database updating will be unavailable. The administrators will be notified about the expiry of the license.

It is essential therefore to review regularly the information in the license key and renew the license in a timely fashion.

4.4.1. Viewing license key information

Information about installed license keys can be reviewed in the logs produced by the *kavscanner, kavmonitor* and *keepup2date* components, since each of them loads the information from the license keys when they are launched.

Moreover, Kaspersky Anti-Virus contains a special *licensemanager* component, which allows you to view more detailed information about the keys together with some analytical data.

All the information may be output to terminal screen.

In order to review information about installed license keys,

enter the following in the command line:

```
licensemanager -s
```

Information about installed licences, similar to the following, will be output to the terminal:

```
Kaspersky license manager Version 5.5

Copyright (C) Kaspersky Lab. 1998-2004.

License file 0003D3EA.key, serial 0038-000419-
0003D3EA, "Kaspersky Anti-Virus for Unix", expires
04-07-2003 in 28 days

License file 0003E3E8.key, serial 011E-000413-
0003E3E8, "Kaspersky Anti-Virus for Linux File Srv
(licence per e-mail address)", expires 25-01-2004 in
234 days
```

In order to review the information about a specific license key,

enter the following in the command line:

```
   licensemanager -k 0003D3EA.key
```

Information similar to the following will be output to the console:

```
Kaspersky license manager Version 5.5
```

```
Copyright (C) Kaspersky Lab. 1998-2004.
Serial 0038-000419-0003D3EA, "Kaspersky Anti-Virus
for Linux", expires 04-07-2003 in 28 days
```

4.4.2. License extension

Extension of your license to use Kaspersky Anti-Virus continues or restores complete application functionality, including continued access to the services listed in section 4.2.5 on p. 28.

The period of license validity depends on the type of licensing that you have selected during application purchase.

In order to extend your license to use Kaspersky Anti-Virus you'll need to:

contact the company from which you purchased the application, and acquire an extension for your license to use Kaspersky Anti-Virus.

or:

extend the license duration directly through Kaspersky Lab by sending an email message to the Sales Department (sales@kaspersky.com), or fill in the appropriate form at the **E-Store→ Renew Your License** section of our site (www.kaspersky.com). After payment you will receive a license key sent to the e-mail address indicated in your order form.

The purchased license key has to be installed. To do so, copy it to the directory assigned for key storage, and specified by the **LicensePath** parameter in the **[path]** section of the configuration file, and restart the computer.

You are advised to update your anti-virus databases after the procedure (please see section 4.1 on p. 18).

Chapter 5. Additional setup

This section describes in detail additional configurations of Kaspersky Anti-Virus functionality. Unlike the required settings made during the installation process (please see Chapter 3 on p. 15) and essential for application functioning, additional setup is performed at the administrator's discretion to extend application functionality and tailor it to fulfill particular business needs.

5.1. Configuring anti-virus protection for file systems

As we have mentioned earlier, computer file systems are protected against viruses by the *kavscanner* component. All the relevant parameters can be subdivided into groups that determine:

- Scan area (see section 5.1.1 on p. 34).
- Object scanning and disinfection mode (see section 5.1.2 on p. 35).
- Operations on objects (see section 5.1.3 on p. 36).
- Parameters for reporting application activity (see section 5.5 on p. 40).

The following sections discuss each of these groups of settings in turn.

5.1.1. Scanning area

The scanning area may be conveniently subdivided into two parts:

- *scanning path* is a list of target directories and objects for virus scanning;
- *scanning objects* means the types of objects which will be scanned for virus presence (archives, etc.).

By default all accessible file system objects are scanned, beginning with the current directory.

Scanning all the computer file systems requires to enter the root first or indicate */* as the scanning area in the command line.

The scanning path can be redefined using the following methods:

- Enumerating directories and files with their absolute or relative (to the current directory) paths, separated by spaces on the command line, when the component starts.
- Defining scanning paths in a text file, with the subsequent command to use the file issued via the **-@ <file_name>** option. Each object in the file is listed on a new line with its absolute path.

  If the command line contains both a scanning path and a text file with a list of objects for scanning, the application will **only** process the objects listed in the file. It will ignore the path specified on the command line.

- Restricting default paths (all beginning with the current directory), or of the paths listed on the command line, can be accomplished by entering masks for files and directories to be excluded from the scanning area, using the parameters **ExcludeMask** and **ExcludeDirs** in the **[scanner.options]** section of the configuration file.
- Disabling *recursive scanning of directories* (**[scanner.options]** section, parameter **Recursion** or **–r** key).
- Creating an alternative configuration file, with a subsequent command to use it issued via the **–c <file_name>** option when the component starts.

The default scanning objects are also specified in the *kav4unix.conf* configuration file (**[scanner.options]** section) and can be redefined:

- directly in that file;
- by command line options when the component is started;
- when an alternative configuration file is used.

5.1.2. Object scanning and disinfection mode

These settings are essential for scanning because they determine whether the application should attempt to cure infected files.

The option to disinfect is **disabled** by default, which means that when scanning the application will only provide notification that viruses, suspicious or corrupted objects have been discovered, by sending a message to the console and to its report (see section 5.5 on p. 40).

As a result of the scanning procedure, each object is assigned one of the following status values:

- **Clean** – no viruses detected (the object is not infected).

- **Infected** – the object is infected.
- **Warning** – the object code resembles a known virus.
- **Suspicious** – the object is possibly infected with an unknown virus.
- **Corrupted** – the object is damaged.
- **Protected** – the object cannot be scanned because it is encrypted (password-protected).

When disinfection mode is enabled (section **[scanner.options]**, parameter **Cure=yes**) only objects with **Infected** status are sent for anti-virus processing. Following disinfection, a file is assigned one of the following status values:

- **Cured** – the object has been successfully disinfected.
- **CureFailed** – object disinfection has failed. These files will be treated according to the rules defined for infected objects.

5.1.3. Operations on suspicious and infected objects

Certain actions may be applied to objects depending upon their status (see section 5.1.2 on p. 35). However, some actions can be defined specifically for files with **Infected**, **Suspicious**, **Warning** or **Corrupted** status, such as:

- *transfer to a specified directory* – relocation of files with a defined status to a *specified* directory, *regular* or *recursive transfer* is possible;
- *removal* of the file from the file system;
- *execution of a certain command* – processing of files using standard Unix commands, scripts, etc.

For **Protected** and **Cured** files the application just outputs its notifications to the console and to the report.

Please note that Kaspersky Anti-Virus distinguishes between a simple object (file) and a compound object (containers consisting of several objects, e.g. archive). The actions to be performed on these two types of objects are also different, and are allocated separate sections in the configuration file. The **[scanner.object]** section is devoted to simple objects, and the section **[scanner.container]** is for compound objects.

Various operations are possible for self-extracting archives: if an archive itself is infected, it is viewed as a simple object, but if archived objects inside it contain viruses, it is treated as a compound one. These two separate operations on the archive are determined by parameters from different sections of the configuration file!

You can select an action to be performed upon specific files using the following methods:

- Specify the actions in the *kav4unix.conf* configuration file if they are to be used as default actions (**[scanner.object]** and **[scanner.container]** sections).

- Indicate the actions in an alternative configuration file and use it when the component starts.

  If no configuration file is indicated in the command line the functional parameters are taken from the default *kav4unix.conf* file.

- Define the actions for the current session using command line options when the *kavscanner* component is started.

The syntax defining actions is similar for simple objects and containers (**[scanner.object]** and **[scanner.container]** sections).

5.2. Setup of real-time anti-virus protection

As discussed in section 4.3 on p. 29, real-time anti-virus protection of computer file system is performed by the *kavmonitor* component.

All parameters for the *kavmonitor* component are stored in the **[monitor.*]** section of the application configuration file.

The *kavmonitor* component is configured to check all objects specified by the user for the presence of virus and malware presence, including:

- packed files;
- archives;
- self-extracting archives;
- e-mail databases;
- plain e-mail messages.

After scanning, the application processes objects using the parameters specified in its configuration file.

By default the cure mode for discovered infected objects is disabled. To enable that option assign the value **Yes** to the **Cure** parameter in section **[monitor.options]** of the application configuration file.

The parameters for the *kavmonitor* component are identical to those for the *kavscanner* component, for which please refer to section 5.1 on p. 34. Let us review some features specific for work with that component.

The scanning area controlled by the *kavmonitor* component can be restricted, by excluding directories and file types using the **ExcludeDirs** and **ExcludeMask** parameters respectively, in the **[monitor.options]** section of the configuration file.

Operations with infected objects performed by *kavmonitor* are identical to the actions of *kavscanner*, except for the fact that *kavmonitor* does not execute Unix commands. Its operations on infected objects are therefore limited to removing them from the file system, or relocating it to a directory specified by the administrator.

5.3. Optimizing Kaspersky Anti-Virus

Kaspersky Anti-Virus offers several methods for optimizing its operations, to decrease CPU load and speed up anti-virus processing of scanned objects.

Use of iChecker™ database, and double-level caching of scanned files.

The application avoids scanning files anew each time they are accessed, by checking to see whether the file has changed since it was last scanned. The algorithm for object (file) scanning for virus presence is as follows:

After initial scanning of any file the information about it (name, checksum) is appended to one of the following databases:

- The iChecker™ database includes information about scanned **clean** files in identified formats, provided by both the *kavmonitor* and *kavscanner* components.
- The scanned files' cache is a database containing information on all the files checked by the *kavmonitor* component. The cache has two levels: the first level contains information about the most frequently accessed **clean** files, and is located in the kernel module, which considerably reduces the time required to access it. If the application finds data on a requested file in the first level cache, it assigns the **Clean** status to that

file and makes no further anti-virus checks. If the first cache level does not contain the required information, the application performs a search in the second level cache which contains data on **all checked files**. Both cache databases exist in RAM, and are not saved when the application shuts down.

Thus, if during the scanning procedure, information about a file is not added to the iChecker™ database, because the file is not clear or has an unsupported format, it is sent to the application cache.

All subsequent user's accesses to a file force a search for the file name in the first cache level and then, if the object has not been found there, in the iChecker™ database and in the second cache level. If the file name is found in any of the databases, its current condition will be compared with the data stored in the database. The file is considered to be unchanged, and is correspondingly not re-scanned, if its current condition is completely identical to the stored data.

If no data can be found for the requested file in either the iChecker database or the cache, the file is scanned for viruses and the databases updated with the results.

CPU load restriction.

Scanning computer file systems may require considerable time if the data volume is large, in which case the CPU load may grow noticeably. As the processor must be able to perform its usual tasks, so it is useful to suspend anti-virus scanning on a computer when a certain load limit is exceeded.

Kaspersky Anti-Virus version 5.5 has such a mechanism. The **MaxLoadAvg** parameter has been added to the **[scanner.options]** section. If the parameter value is specified, *kavscanner* checks the current CPU load value (**load average**) before scanning each new file. If the value exceeds the figure specified in the configuration file, the scanner will suspend its work until the **load average** value decreases to the specified threshold.

5.4. Localization of displayed date and time formats

While working, Kaspersky Anti-Virus compiles reports for each of its components, and notifications for users and administrators, which are always supplemented with the date and time at which they occurred.

By default, Kaspersky Anti-Virus uses time and date formats conforming to those used by the C function strftime:

%H:%M:%S – displayed time format.

%d/%m/%y – displayed date format.

An administrator may change the date and time format through parameters in the **[locale]** section of the configuration file. Example of possible formats include:

%I:%M:%S %P – for time output in twelve-hour format (**TimeFormat** parameter) with an am/pm indication.

%y/%m/%d and **%m/%d/%y** – for date output (**DateFormat** parameter) in the year/month/date or month/date/year formats respectively.

5.5. Reporting parameters in Kaspersky Anti-Virus

Results of operations performed by all Kaspersky Anti-Virus‘s components are summarized in a report output to a log file.

Results of anti-virus processing of computer file systems are also output to the console. By default the information output to a report and to the console is identical. Additional configuration is needed to display different information on the console from that in the report log.

The amount of output information can be altered by changing the *report detail level*.

The **level of detail** is a number that sets the level of verboseness for information regarding the components’ work. Each subsequent level (higher number) includes information of the previous level, together with some additional data.

The possible levels of report details are listed in the table below.

Levels	Level name	Meaning
	Fatal errors	Only information regarding critical errors, which terminate the program due to impossibility of executing an action. For example: the application component is infected; or a vital action failed, such as scanning, database loading, or license key loading.

Levels	Level name	Meaning
1	Errors	Information about other errors, including those not causing components to terminate, e.g. information regarding a file scanning failure.
2	Warning	Information about errors, which can cause termination of product operation (e.g., information about insufficient free disk space).
3	Info, Notice	Important information messages, e.g. whether the component is running or not, the path to the configuration file, scan area, information regarding anti-virus databases, license keys, and the resulting statistics.
4	Activity	Messages regarding object scanning, according to the level of detail set for the scanning report.
10	Debug	All debug messages, for example, configuration file contents.

Information regarding fatal errors in component operation is output regardless of the selected level of detail. The optimal level of detail is **4**, which is set by default.

Chapter 6. Uninstalling Kaspersky Anti-Virus

The procedure for uninstalling Kaspersky Anti-Virus requires the following:

- superuser privileges (**root**). If you do not have these privileges when initiating the uninstall procedure, you will have to log into the system as **root** user.
- Installation log file.
- Names and sizes of the files installed as parts of Kaspersky Anti-Virus must be exactly the same as specified in the installation log file.

If you installed Kaspersky Anti-Virus using its .rpm package enter the following in the command line to begin the uninstall procedure:

```
rpm -e <package_name>
```

If you installed Kaspersky Anti-Virus using its .deb package enter the following in the command line to begin the uninstall procedure:

```
dpkg -r <package_name>
```

If you installed Kaspersky Anti-Virus using its .tar.gz package enter the following in the command line to begin the uninstall procedure:

```
uninstall.pl
```

If you installed Kaspersky Anti-Virus using its .pkg package enter the following in the command line to begin the uninstall procedure:

```
pkg-delete <package_name>
```

The program will be uninstalled automatically. A notification message will be output to the console as soon as the procedure is over.

Chapter 7. Testing the operation of Kaspersky Anti-Virus

After installing and adjusting Kaspersky Anti-Virus, you are advised to test the correctness of its settings and operation using a series of test "viruses".

The test virus was specially designed by the European Institute for Computer Antivirus Research organization, eicar for testing anti-virus products.

The test "virus" IS NOT ACTUALLY A VIRUS because it does not contain code that can really harm your computer. However, most anti-virus products identify this file as a virus.

Never use real viruses for testing the operation of an anti-virus product!

You can download the test "virus" from the official website of the **EICAR** organization at: http://www.eicar.org/anti_virus_test_file.htm. If you have no Internet connection, you can create your own test "virus". To create a test "virus", type the following string in any text editor and save the file as **eicar.com**:

```
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-
FILE!$H+H*
```

The file downloaded from the **EICAR** website, or created as described above, contains the body of a standard test "virus". The application will detect it, assign the **Infected** status to it and apply the action defined by the administrator for handling objects of this status.

To test the response of the application to other types of objects, modify the body of this standard test "virus" by adding one of the prefixes (please see the table below).

Table. Test "virus" modifications

Prefix	Object type
No prefix, standard test "virus"	**Infected**. The object cannot be disinfected.
CORR–	**Corrupted**.

Prefix	Object type
SUSP–	**Suspicious** (unknown viral code).
WARN–	**Warning** (modified code of a known virus).
ERRO–	**Error while scanning the object**.
CURE–	**Cured**. The object is disinfected; the text of the "virus" body is changed for CURE.
DELE–	The object is automatically deleted.

The first table column lists prefixes to be added at the beginning of the string of the standard test "virus" (for example, CORR–X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*). The second column of this table contains the status assigned by the application after the prefix has been added. The actions for each status of object are defined by anti-virus application settings customized by the administrator.

Chapter 8. What's new in versions 5.0 and 5.5 compared to version 4.0

Apart from modifications and improvements, the current version of the application has these new features compared to Kaspersky Anti-Virus 4.0:

- All the application components have been transferred to a new anti-virus engine, which puts a lower load on the processor but provides the same functionality.
- A number of drawbacks of Kaspersky Anti-Virus for Unix v. 4.0 related to its architectural features have been corrected, including some major security bug-fixes. Therefore we recommend that all users of version 4.0 migrate to using version 5.5.
- The web-based WebTuner configuration component has been replaced with a plug-in module for the commonly used Webmin application. This has extended the functionality of the application.
- Configuration files of all the components have been merged into one *kav4unix.conf* file, thus making it possible to manage the entire application. Additionally, the configuration of the entire application has been transferred from a binary to a more convenient text-based framework.
- The time and date formats, displayed in the application's operation reports and in its notifications, can now be configured.
- It is now possible to use script files to process objects with different statuses (infected, corrupted, etc.).
- The application is distributed in the form of standard installation packages for the supported operating system distributions (rpm, deb, tgz).
- The application licensing policy has been changed:
    - After the license expires, Kaspersky Anti-Virus remains completely functional except for the feature for updating of the anti-virus databases.
    - Kaspersky Anti-Virus does not work without a license key.
- The *kavucc* (Control Centre) component has been excluded from the package due to simplification of the application architecture.

- The *kavinspector* (Inspector), *kavtuner* (Tuner) and *kavdaemon* (Daemon) components have been excluded from the application package.

Appendix A. Frequently asked questions

This chapter is devoted to the most frequently asked users' questions pertaining to installation, setup and operation of Kaspersky Anti-Virus; here we shall try to answer them in detail.

Question: Is it possible to use Kaspersky Anti-Virus with anti-virus products of other vendors?

We recommend uninstalling anti-virus products of other vendors prior to installation of Kaspersky Anti-Virus to avoid software conflicts.

Question: Kaspersky Anti-Virus does not rescan a file. Why?

Kaspersky Anti-Virus does not rescan files which have not changed since their last scan.

That has become possible due to new iChecker and iStreams technologies. The application implements the technology using a database of file checksums and file checksum storage in alternate NTFS streams.

Question: Why does Kaspersky Anti-Virus cause a certain decrease in computer performance, noticeably loading the CPU?

Virus detection is a computationally intensive mathematical problem requiring structural analysis, checksum calculation and mathematical data conversions. Processor time is therefore the main resource consumed by the Anti-Virus, and each new virus added to the anti-virus database increases the overall scanning time.

Other anti-virus products speed up scanning by excluding from their databases both viruses which are less easily detectable or less frequent (e.g. in a specific geographic location), and file formats that require complicated analysis (e.g., PDF). Kaspersky Lab believes that the purpose of anti-virus protection is to establish real and complete anti-virus security for its users.

Experienced users can, of course, accelerate anti-virus scanning by disabling scanning of various file types. However, please keep in mind that it will decrease the overall security level.

Kaspersky Anti-Virus recognizes more than 700 formats of archived and packed files. This is essential for anti-virus security because harmful executable code may be hidden inside files of any recognized format. However, despite the daily growth in the number of viruses detected by Kaspersky Anti-Virus (approximately 30 new viruses appear daily) as well as the ever increasing number of recognized file formats, this new version of our application functions faster than previous ones. That is achieved through the use of new unique technologies, such as iChecker™ and iStreams™, developed at Kaspersky Lab.

Question: Why do I need the key file? Will my copy of the Anti-Virus work without it?

No, Kaspersky Anti-Virus does not work without a license key.

If you are still deciding whether or not to purchase Kaspersky Anti-Virus, we can provide you with a temporary key file (trial key), which will only work either for two weeks or for a month. When this period expires, the key will be blocked.

Question: What happens when the license expires?

After expiration of the license, Kaspersky Anti-Virus will continue operating, but anti-virus database updating will be disabled. The anti-virus application will continue cleaning infected objects but it will be using the old anti-virus databases.

If such a situation arises, contact either the company from which you purchased Kaspersky Anti-Virus, or Kaspersky Lab directly, for license extension.

Question: license key for Kaspersky Anti-Virus is on a floppy disk. What should I do if my computer has no floppy drive?

There are several solutions for the problem.

You can write an e-mail with problem description to the Sales Department of Kaspersky Lab (sales@kaspersky.com). Be sure to mention the date and location where you have purchased Kaspersky Anti-Virus and its complete registration number. Sales department staff will send your key file to the e-mail address, which you have specified.

You can also read the floppy disk contents on another computer equipped with a floppy drive and then record the data to another medium, which your computer can read. While installing Kaspersky Anti-Virus, specify that medium as the source of your license key.

Alternatively, you can read the floppy disk contents on another computer with a corresponding drive and e-mail the key file to your own address. Receive the letter, save its data attachment in any folder on your hard drive and specify the folder as the license key source while installing Kaspersky Anti-Virus.

Question: My Anti-Virus does not work.

What should I do?

First, check if a solution for your problem is provided in this documentation, especially in this section or at our web site.

In addition, we recommend that you apply for support to the distributor from whom you purchased Kaspersky Anti-Virus, or write to Kaspersky Lab's Technical Support (support@kaspersky.com) or to the address included in the license key information.

To make sure your request is answered as soon as possible, follow these suggestions:

1. In the message header, specify the operating system of your computer, the name of the component you are experiencing problems with, and briefly describe the problem. For example: **Linux, Webmin, no access to settings of the licensed users' list**.
2. Compose your messages in plain text format.
3. At the beginning of the message, specify the exact versions of the operating system and Kaspersky Anti-Virus distribution package and provide the number of your license.
4. Clearly describe the problem in brief. Keep in mind that, when reading your mail, the support service officers do not yet know about your problem. They can only help after fully understanding and reproducing it.
5. Send the following data, packed into one archive, to the Technical support service:
   - files from the */etc/kav/* directory;
   - logs of the Anti-Virus components, e.g., /var/log/*aveserver.log*;
   - information output to console by the **ps -ax** command;
   - your license key file.
6. Make sure to specify in your mail if your computer system contains any of the following:

- SCSI controller;
- a very old or very new processor, or more than one processor;
- less than 64 MB or more than 2 GB of RAM.

Question: What are the daily updates for?

A few years ago viruses were transmitted on floppy disks, and adequate computer protection could be achieved by installation of an anti-virus program followed by infrequent updates to its anti-virus database. However, recent virus epidemics spread around the world in a matter ofl hours, and anti-virus protection with old databases may be helpless against a new threat. In order to resist new viruses, you should update the anti-virus databases daily.

Every year Kaspersky Lab increases the frequency of its updates issued for the anti-virus databases. Currently they are updated every three hours.

Updating of the application modules is an additional feature that allows both correction of discovered vulnerabilities and addition of new functions.

Question: What are the changes to the updating service of version 5.0?

The Kaspersky Lab 5.0 product suite features a new updating service, which has been developed in accordance with the requests of our users. It automates the whole updating procedure, from the preparation of updates in Kaspersky Lab to the moment that relevant files are updated on clients' computers.

Advantages of the new updating service include:

- *Ability to resume downloading of files after disconnection.* Upon reconnection only files which have not been downloaded are retrieved.
- *Cumulative updates are now half the size*. A cumulative update contains the whole anti-virus database; therefore its size exceeds considerably the size of typical updates. The new service employs a special technology, which allows using an already existing anti-virus database for a cumulative update.
- *Accelerated downloading from the Internet*. Kaspersky Anti-Virus picks up a Kaspersky Lab's updates server located in your region. Furthermore, servers are allocated according to their

performance, so you will not be sent to an overloaded server while there is another idle server available.

- *Use of key black lists*. Unlicensed and illegal users are now prevented from using the updating service. Licensed users therefore do not suffer from inability to contact overloaded updates' servers.
- *Corporate enterprises can now create a local updates' server*. This feature is designed for organizations where a single LAN unites computers protected by Kaspersky Lab products. Any computer on the LAN can be turned into an updates' server that retrieves updates from the Internet and shares them with the other networked computers.

Question: is it possible for an intruder to replace the anti-virus database?

Every anti-virus database has a unique signature checked by Kaspersky Anti-Virus when accessing the database. If the signature is wrong or the date of the database is later than that of the license expiration, Kaspersky Anti-Virus will not use it.

Question: will Kaspersky Anti-Virus work with my Linux distribution?

Version 5.5 of Kaspersky Anti-Virus has been tested with RedHat, Debian and SuSE distributions and Kaspersky Anti-Virus packages have been compiled specifically for the listed distributions.

Please see the supported OS versions in section 1.3 on p. 6.

If your distribution is 100 percent compatible with a supported one (for example, ASPLinux is compatible with Red Hat Linux), then the probability of critical problems is very low.

Users of distributions that are not included in the list supported by Kaspersky Lab may experience incorrect application operation, which will be caused by specific details of the operating system. For example, your OS distribution may use a different version of a library or its system initialization scripts may have a non-standard location. In such cases Kaspersky Lab's Technical Support service will be unable to help you.

Question: how can I decompress a .tgz or .tar.gz archive?

.tgz or *.tar.gz* archives are unpacked using the following command:

```
tar -zxvf <archive_name>
```

Please consult man pages for the **tar** program for details.

Question: why does kavmonitor start several simultaneous processes?

The number of running *kavmonitor* processes is determined by the **CheckFileLimit** parameter in the application configuration file; it specifies the number of files processed simultaneously. Therefore the number of monitor processes is always greater than 1 (the monitor runs 20 processes by default). If there are no files for scanning, the processes do not consume any system resources.

Question: can I control Kaspersky Anti-Virus using the Network Control Centre for Windows?

Network Control Centre for Windows cannot be used for operations with Kaspersky Anti-Virus for Linux/FreeBSD/OpenBSD Workstations and File Servers. In this version of the product we have provided for an opportunity to control the software remotely via a special Webmin module.

Question: how do I save software console output to a file?

In order to save the information output by Kaspersky Anti-Virus to the console during work you should either add corresponding settings to the configuration file or enter the following in the command line:

```
$ some_app > ./text_file 2>&1
```

where:

`some_app` – means the software, the standard output and error messages of which you would like to have saved to a file;

`text_file` – full path to the file, where the information will be recorded.

E.g.:

```
$keepup2date > ./updater.log 2>&1
```

In that case, standard output messages as well as error messages from the *keepup2date* component will be output to the *updater.log* file in the current directory.

Appendix B. Malware in Unix environment

Viruses are much less frequent in Unix systems than, for example, under Windows due to some peculiarities of those platforms. Trojan horses and network worms are less rare.

Malicious programs spread through networks using various ways including the exploitation of software "holes". This appendix reviews in more detail the types of malware existing under UNIX and the methods they use for system infection.

B.1. Viruses

A virus is a program (certain executable code and/or instructions), capable of reproducing its copies (which do not have to be totally identical to the original) and embedding them in different objects and/or resources of computer systems, networks, etc. without the user's knowledge. The copies also have the ability to spread further.

A study of a virus environment reveals that viruses in Unix are generally of the file type, and either attach their code to executable files or create phantom files.

The following subclasses are defined according to their functioning algorithm:

- *memory-resident (TSR) virus* means a virus which leaves its resident part in system RAM after infection, subsequently intercepting system calls to infectable objects and incorporating itself into them. Resident viruses stay in memory and remain active until computer power-down or an operating system reboot.
- *non-resident virus* is a virus which does not infect computer memory and remains active for a limited time. Some viruses leave small resident programs in memory, which do not spread the virus.

As a rule viruses in Unix are not dangerous – their influence is limited to decreasing free disk space, and temporary graphics, sound and other effects. Some of them are completely harmless since they do not alter computer functionality in any way, except for decreasing free disk space due to their existence in the file system.

Let us review some examples of Unix viruses:

ELF_SNOOPY is a virus infecting executable Unix files.

Virus algorithm: it finds all executable files present on a workstation, renames them to files with an .X23 extension and moves to a created

/E directory. Then the virus copies itself to the original files and changes their attributes to **777**. It also creates a user **snoopy** with rights 777 in the main password list of the infected workstation.

Linux.Bliss is a group of non-resident viruses, written in GNU C and in ELF format, which infects Linux executables.

Virus algorithm: when launched, the virus searches for executable files on a workstation and infects them by inserting itself at the start of the file, appending an identification string to the end of file. Virus activity is limited by the rights of the user who started it (only accessible files are infected). If a user has system privileges the virus may spread to the whole computer.

Linux.Diesel is a harmless non-resident Linux virus infecting Linux executables.

Virus algorithm: after start the virus reads its code from the carrier file, searches for Linux executables in system subdirectories and writes itself into the middle of each file, thus increasing the size of the last section.

Linux.Siilov is a harmless non-resident ELF-format Linux virus infecting Linux executables.

Virus algorithm: it uses two methods of file infection: resident and non-resident. In the resident method, the virus remains in system memory and infects files in the background. In the non-resident method, the virus searches for executable files on disk and infects them.

Linux.Winter is a harmless non-resident Linux virus. It is very small – just 341 bytes.

Virus algorithm: after start the virus receives control, searches for ELF files (Linux executables) in the current directory and infects them.

B.2. Trojan software

A trojan horse is a program which performs unauthorized actions. When launched, a trojan horse installs itself in the system and then begins monitoring it; the user receives no notifications about the trojan’s actions on the system. The computer is open for remote control.

Trojan software spreads through networks.

One typical representative of the trojan family in Unix is **TROJ_IRCKILL** – it is actually a collection of software tools to disconnect users from IRC channels. The collection integrates four attack utilities: FLOOD, MCB (Multiple Collide BOTs), SUMO BOTs and FLASH – a special “flood" type for use in Unix.

The FLASH attack type is used for direct modem disconnection by sending a **ping** command with "incorrect" data in a certain sequence to a certain IP address. The user's modem will interpret the data as a command to disconnect and the user will be disconnected from the Internet. However, this type of attack only works with some modem types.

MCB attacks are performed via IRC channels. At times, when IRC servers are unable to synchronize with each other (net split) the trojan imitates a connection, duplicating the user's name (nickname). After IRC servers achieve synchronization the user name becomes invalid and the user gets disconnected from the IRC channel.

Attacks by FLOOD BOTS/SUMO BOTS are also used in IRC networks, "producing" numerous users with random nicknames. The attack is used to "flood" an IRC channel or a user, who sends or receives chat messages until the user's computer reaches its bandwidth limit. Then the user will also be disconnected from the IRC channel.

Root kit is a collection of tools used by hackers in order to receive root access to a remote computer. It uses standard Unix programs – ps and ls. The only efficient method of recovery for computers hacked using the Root kit is to completely delete hard disk contents, reinstall the operating system and restore important data from a regular backup copy,.

B.3. Network worms

A network worm is a malicious program which does not add itself to executable objects, but copies itself to network resources instead. The group is named by analogy with the ability of worms to "crawl" through networks and other informational channels.

They penetrate computer memory from computer networks, calculate network addresses of other computers and send their own copies to those addresses.

Worms may sometimes have work files on system disks, but some use no resources, except RAM, on a computer at all.

Worm.Linux.Ramen was the first known worm infecting RedHat Linux systems. It infects remote RedHat Linux systems exploiting the buffer overflow problem. The software "hole" allows sending executable code to a remote computer and its execution there, unnoticed by the administrator or user.

Infection source: a .**tgz** archive from a network.

Algorithm: using the buffer overrun problem the worm sends a short portion of code to a remote computer. When the main worm component (*start.sh* file) starts, it opens a connection successively downloading other components, which determine the addresses of the systems being attacked, and use a buffer overrun breach to send each one a worm loader, which in its turn completes loading and starts the main portion of the worm code. The main page of a web server is replaced with an

HTML file containing the following text: "RameN Crew – Hackers looooooooooooove noodles". Finally the worm sends an e-mail message to two addresses, restarts the system and begins scanning the Internet again.

The worm also adds a command for starting its main file to the */etc/rc.d/rc.sysinit* system initialization file. As a result the worm is started during all subsequent reboots of an infected system.

Worm.Linux.Lion is an Internet worm attacking Linux servers. It uses a security breach in the BIND DNS service to penetrate computer systems.

Algorithm: the worm scans the Internet searching for systems with a root access vulnerability. When it discovers such a system, the worm infects it, collects information on it (IP address, logins, passwords) in the *mail.log* file and then sends it to the e-mail address *1i0nsniffer@china.com*.

In addition, the worm attempts to contact the www.51.net web site, registered in China, via the Internet and download the file *crew.tgz* from it. The archive is then uncompressed on the infected computer, and routines are installed which make the infected system in turn scan global network resources in search of new victims.

mIRC.Acoragil and **mIRC.Simpsalapim** were the first known mIRC worms. Their names originate from the code words used by the worms: if the text sent to a channel by any user contains the *Acoragil* line, then all users infected with the **mIRC.Acoragil** worm will be automatically disconnected from that channel. The **mIRC.Simpsalapim** worm reacts in a similar manner to the *Simpsalapim* line.

Infection source: through the network, using mIRC commands, the worms send their code in *SCRIPT.INI* file to each new user connecting to the channel.

Algorithm: the worms contain a trojan code portion. **mIRC.Simpsalapim** can capture an IRC channel: if the mIRC channel owner is infected then input of the *ananas* code word will enable an intruder to seize control of the channel.

mIRC.Acoragil sends DOS, Windows or Unix system files according to received code words. Some code words are chosen so as to attract no attention of the victim – *hi* or *the*. One of the worm's modifications sends the Unix passwords file to an intruder.

Worm.Linux.Adm is an Internet worm infecting Linux systems. The worm sends a small portion of its code to remote computers, runs it there, downloads its main portion and then executes it,.

Infection source: via networks; the worm infects remote Linux systems by sending itself using a buffer overflow breach. The hole allows sending executable code to a remote computer and its execution there, unnoticed by the administrator or user.

Appendix C. 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 email 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 and Poland. 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 250 specialists, each of whom is proficient in anti-virus technologies, with 9 of them holding M.B.A. degrees, 15 holding Ph.Ds, and two 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. At all times, the company’s products 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 and Internet-gateways, 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. Kaspersky Lab's anti-virus database is updated every 3 hours. The company provides its customers with a 24-hour technical support service, which is available in several languages to accommodate its international clientele.

C.1. Other Kaspersky Lab Products

Kaspersky Anti-Virus Personal

Kaspersky Anti-Virus Personal protects home computers running Windows 98/ME/2000/NT/XP from all types of known viruses, including Riskware. The application constantly monitors all possible sources of virus penetration, including email, Internet, floppy disks, and CDs. Unknown viruses are efficiently detected and processed by a unique heuristic data analysis system. The two distinct modes of the application's operation (that can be used either separately or jointly) are:

- **Real-Time Protection** – anti-virus scan of all files being run, opened or saved on the protected computer.
- **On-Demand Scan** – scanning and disinfection of the entire computer or individual disks, files or folders. You can launch such a scan manually using the graphical interface, or schedule a regular automated scan.

Kaspersky Anti-Virus Personal does not rescan objects which have not been modified since their previous scan. This rule now applies both to real-time protection and to the on-demand scan. This feature **greatly improves the speed and performance of the application**.

Kaspersky Anti-Virus Personal provides reliable protection against viruses that attempt to penetrate computers via email messages. The application automatically scans and disinfects all incoming (POP3) and outgoing (SMTP) email messages and efficiently detects viruses in email databases.

Kaspersky Anti-Virus Personal supports over 700 formats of archived and compressed files and ensures automatic anti-virus scanning of their content, and removal of malicious code from files within **ZIP**, **CAB**, **RAR** and **ARJ** archives.

The application's settings can easily be adjusted by selecting one of three pre-defined levels: **Maximum Protection**, **Recommended Protection** and **Maximum Speed**.

The anti-virus database is updated every three hours. Database delivery is guaranteed even if the internet connection is interrupted or switched during the download process.

Kaspersky Anti-Virus Personal Pro

This package has been designed to deliver comprehensive anti-virus protection to home computers running Windows 98/ME/2000/NT/XP as well as Microsoft Office 2000 applications. Kaspersky Anti-Virus Personal Pro includes an easy-to-use application for automatic retrieval of daily updates to the anti-virus database and application modules. A second-generation heuristic analyzer efficiently detects unknown viruses. Kaspersky Anti-Virus Personal includes many interface enhancements, making it easier than ever to use the application.

Kaspersky Anti-Virus Personal Pro has the following features:

- **On-demand scan** of local disks;
- **Real-time automatic protection** of all accessed files from viruses;
- **Mail filter** automatically scans and disinfects all incoming and outgoing mail traffic (POP3 and SMTP) and effectively detects viruses in mail databases;
- **Behavior blocker** that provides maximum protection of MS Office applications from viruses;
- **Archive scans –** Kaspersky Anti-Virus recognizes over 700 formats of archived and compressed files and ensures automatic anti-virus scanning of their content and removal of malicious code from files within **ZIP**, **CAB**, **RAR** and **ARJ** archives.

Kaspersky Anti-Hacker

Kaspersky Anti-Hacker is a personal firewall that is designed to safeguard computers running any Windows operating system. It protects your computer against unauthorized access and external hacker attacks from either the Internet or the local network.

Kaspersky Anti-Hacker monitors the TCP/IP network activity of all applications running on your machine. When it detects a suspicious action, it prevents the suspicious application from accessing the network. This enhances your privacy and provides 100% security for confidential data stored on your computer.

The application's SmartStealth™ technology prevents hackers from detecting your computer from the outside. In this stealthy mode, the application works seamlessly to keep your computer protected while you are on the Web. The application provides conventional transparency and accessibility of information.

- Kaspersky Anti-Hacker also blocks most common network hacker attacks and monitors for attempts to scan computer ports.

- Configuration of the application is simply a matter of choosing one of five security levels. By default, the application starts in self-learning mode, which will automatically configure your security system depending on your responses to various events. This makes the firewall adjustable to your specific preferences and your particular needs.

Kaspersky Security for PDA

Kaspersky Security for PDA provides reliable anti-virus protection of data stored on PDAs running Palm OS or Windows CE. It also offers anti-virus protection from corrupted files transferred from a PC or an extension card, from ROM files, or from databases. This software package includes an optimal combination of the following anti-virus tools:

- **anti-virus scanner** to scan the data stored on both the PDA and extension card on demand;
- **anti-virus monitor** to intercept viruses in files that are either copied from other handhelds or are transferred using HotSync™ technology.

Kaspersky Security for PDA protects your handheld (PDA) from unauthorized intrusion by encrypting both access to the device and data stored on memory cards.

Kaspersky Anti-Virus Business Optimal

This package provides a configurable security solution for small- and medium-sized corporate networks.

Kaspersky Anti-Virus Business Optimal includes full-scale anti-virus protection[1] for:

- *Workstations* running Windows 98/ME/NT/2000/XP Workstation, and Linux;
- *File and application servers* running Windows NT 4.0 Server, Windows 2000, 2003 Server/Advanced Server, Windows 2003 Server, Novell Netware, FreeBSD and OpenBSD, and Linux;
- *Email clients*, namely Microsoft Exchange 5.5/2000/2003, Lotus Notes/Domino, Postfix, Exim, Sendmail, and Qmail;
- *Internet-gateways*: CheckPoint Firewall –1; Microsoft ISA Server.

The Kaspersky Anti-Virus Business Optimal distribution kit includes Kaspersky Administration Kit, a *unique tool for automated deployment and administration*.

All of these components are interoperable so that any of them can be selected, according to the operating systems and applications you use.

Kaspersky Corporate Suite

This package provides corporate networks of any size and complexity with comprehensive, scalable anti-virus protection. The package components have been developed to protect every tier of a corporate network, even in mixed computer environments. Kaspersky Corporate Suite supports the majority of operating systems and applications installed across an enterprise. All package components are managed from one console and have a unified user interface.

[1] Depending on the type of distribution kit.

Kaspersky Corporate Suite delivers a reliable, high-performance protection system that is fully compatible with the specific needs of your network configuration.

Kaspersky Corporate Suite provides comprehensive anti-virus protection for:

- *Workstations* running Windows 98/ME/NT/2000/XP, and Linux;
- *File and application servers* running Windows NT 4.0 Server, Windows 2000, 2003 Server/Advanced Server, Novell Netware, FreeBSD, OpenBSD and Linux;
- *Email clients*, including Microsoft Exchange Server 5.5/2000/2003, Lotus Notes/Domino, Sendmail, Postfix, Exim and Qmail;
- *Internet-gateways*: CheckPoint Firewall –1; Microsoft ISA Server;
- *Hand-held computers* (PDAs), running Windows CE and Palm OS.

The Kaspersky Corporate Suite distribution kit includes Kaspersky Administration Kit, a *unique tool for automated deployment and administration*.

All of these components are fully interoperable so that any of them can be chosen, according to the operating systems and applications you use.

Kaspersky Anti-Spam

Kaspersky Anti-Spam is a cutting-edge software suite that is designed to help organizations with small- and medium-sized networks wage war against the onslaught of undesired email (spam). The application combines the revolutionary technology of linguistic analysis with modern methods of email filtration, including RBL 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, Kaspersky Anti-Spam monitors incoming email and acts as a barrier to unsolicited email. The application 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 with samples provided by Kaspersky Lab’s linguistic laboratory.

Kaspersky Anti-Spam Personal

Kaspersky Anti-Spam Personal is designed to protect users of mail client programs Microsoft Outlook and Microsoft Outlook Express against unwanted email messages (spam).

Kaspersky Anti-Spam Personal software package is a powerful tool that detects spam in incoming email messages received via the POP3 or IMAP4 protocols (only for Microsoft Outlook).

The filtering process involves the analysis of all attributes of the message (sender's and recipient's addresses and headers), content filtration (analysis of the content of the letter, both the subject and any attached files), using unique linguistic and heuristic algorithms.

The application's performance is enhanced by daily updates to the content filtration database with samples provided by Kaspersky Lab's linguistic laboratory.

C.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 application by phone or via email. 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
General information	WWW: http://www.kaspersky.com http://www.viruslist.com Email: sales@kaspersky.com

Appendix D. License agreement

Standard End User License Agreement

> NOTICE TO ALL USERS: CAREFULLY READ THE FOLLOWING LEGAL AGREEMENT ("AGREEMENT") FOR THE LICENSE OF SPECIFIED SOFTWARE ("SOFTWARE") PRODUCED BY KASPERSKY LAB. ("KASPERSKY LAB").
>
> IF YOU HAVE PURCHASED THIS SOFTWARE VIA THE INTERNET BY CLICKING THE ACCEPT BUTTON, YOU (EITHER AN INDIVIDUAL OR A SINGLE LEGAL ENTITY) CONSENT TO BE BOUND BY AND BECOME PARTY TO THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE BUTTON THAT INDICATES THAT YOU DO NOT ACCEPT THE TERMS OF THIS AGREEMENT, AND DO NOT INSTALL THE SOFTWARE.
>
> IF YOU HAVE PURCHASED THIS SOFTWARE ON A PHYSICAL MEDIUM, HAVING BROKEN THE CD'S SLEEVE, YOU (EITHER AN INDIVIDUAL OR A SINGLE ENTITY) ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT DO NOT BREAK THE CD's SLEEVE ,DOWNLOAD, INSTALL OR USE THIS SOFTWARE. IF YOU HAVE BROKEN THE CD'S SLEEVE OR OPENED THE BOX, YOU WILL NOT BE ENTITLED TO RETURN THE SOFTWARE FOR REFUND. SOFTWARE FOR HOME USE (KASPERSKY ANTI-VIRUS PERSONAL, KASPERSKY ANTI-VIRUS PERSONAL PRO, KASPERSKY ANTI-HACKER, KASPERSKY SECURITY FOR PDA) PURCHASED AS A DOWNLOAD VIA THE INTERNET MAY BE RETURNED FOR A FULL REFUND WITHIN 14 DAYS AFTER PURCHASE FROM KASPERSKY LAB, IT'S AUTHORIZED DISTRIBUTOR OR RESELLER. OTHER PRODUCTS ARE NON REFUNDABLE. THE RIGHT TO RETURN AND REFUND EXTENDS ONLY TO THE ORIGINAL PURCHASER.

All references to "Software" herein shall be deemed to include the software activation key ("Key Identification File") with which you will be provided by Kaspersky Lab as part of the Software.

1. License Grant. Subject to the payment of the applicable license fees, and subject to the terms and conditions of this Agreement, Kaspersky Lab hereby grants you the non-exclusive, non-transferable right to use one copy of the specified version of the Software and the accompanying documentation (the "Documentation") for the term of this Agreement solely for your own internal business purposes. You may install one copy of the Software on one computer, workstation, personal digital assistant, or other electronic device for which the Software was designed (each a "Client Device"). If the Software is licensed as a suite or bundle with more than one specified Software product, this license

applies to all such specified Software products, subject to any restrictions or usage terms specified on the applicable price list or product packaging that apply to any such Software products individually.

1.1 Use. The Software is licensed as a single product; it may not be used on more than one Client Device or by more than one user at a time, except as set forth in this Section.

1.1.1 The Software is "in use" on a Client Device when it is loaded into the temporary memory (i.e., random-access memory or RAM) or installed into the permanent memory (e.g., hard disk, CD-ROM, or other storage device) of that Client Device. This license authorizes you to make only as many back-up copies of the Software as are necessary for its lawful use and solely for back-up purposes, provided that all such copies contain all of the Software's proprietary notices. You shall maintain records of the number and location of all copies of the Software and Documentation and will take all reasonable precautions to protect the Software from unauthorised copying or use.

1.1.2 If you sell the Client Device on which the Software is installed, you will ensure that all copies of the Software have been previously deleted.

1.1.3 You shall not decompile, reverse engineer, disassemble or otherwise reduce any part of this Software to a humanly readable form nor permit any third party to do so. The interface information necessary to achieve interoperability of the Software with independently created computer programs will be provided by Kaspersky Lab by request on payment of its reasonable costs and expenses for procuring and supplying such information. In the event that Kaspersky Lab notifies you that it does not intend to make such information available for any reason, including (without limitation) costs, you shall be permitted to take such steps to achieve interoperability, provided that you only reverse engineer or decompile the Software to the extent permitted by law.

1.1.4 You shall not make error corrections to, or otherwise modify, adapt, or translate the Software, nor create derivative works of the Software, nor permit any third party to copy the Software (other than as expressly permitted herein).

1.1.5 You shall not rent, lease or lend the Software to any other person, nor transfer or sub-license your license rights to any other person.

1.1.6 You shall not use this Software in automatic, semi-automatic or manual tools designed to create virus signatures, virus detection routines, any other data or code for detecting malicious code or data.

1.2 Server-Mode Use. You may use the Software on a Client Device or on a server ("Server") within a multi-user or networked environment ("Server-Mode") only if such use is permitted in the applicable price list or product packaging for the Software. A separate license is required for each Client Device or "seat" that may connect to the Server at any time, regardless of whether such licensed Client Devices or seats are concurrently connected to or actually accessing or using the Software. Use of software or hardware that reduces the number of

Client Devices or seats directly accessing or utilizing the Software (e.g., "multiplexing" or "pooling" software or hardware) does not reduce the number of licenses required (i.e., the required number of licenses would equal the number of distinct inputs to the multiplexing or pooling software or hardware "front end"). If the number of Client Devices or seats that can connect to the Software exceeds the number of licenses you have obtained, then you must have a reasonable mechanism in place to ensure that your use of the Software does not exceed the use limits specified for the license you have obtained. This license authorises you to make or download such copies of the Documentation for each Client Device or seat that is licensed as are necessary for its lawful use, provided that each such copy contains all of the Documentation’s proprietary notices.

1.3 Volume Licenses. If the Software is licensed with volume license terms specified in the applicable product invoicing or packaging for the Software, you may make, use or install as many additional copies of the Software on the number of Client Devices as the volume license terms specify. You must have reasonable mechanisms in place to ensure that the number of Client Devices on which the Software has been installed does not exceed the number of licenses you have obtained. This license authorizes you to make or download one copy of the Documentation for each additional copy authorized by the volume license, provided that each such copy contains all of the Document's proprietary notices.

2. Duration. This Agreement is effective for one (1) year unless and until earlier terminated as set forth herein. This Agreement will terminate automatically if you fail to comply with any of the conditions, limitations or other requirements described herein. Upon any termination or expiration of this Agreement, you must immediately destroy all copies of the Software and the Documentation. You may terminate this Agreement at any point by destroying all copies of the Software and the Documentation.

3. Support.

(i) Kaspersky Lab will provide you with the support services ("Support Services") as defined below for a period of one year following:

(a) Payment of its then current support charge, and:

(b) Successful completion of the Support Services Subscription Form as provided to you with this Agreement or as available on the Kaspersky Lab website, which will require you to produce the Key Identification File which will have been provided to you by Kaspersky Lab with this Agreement. It shall be at the absolute discretion of Kaspersky Lab whether or not you have satisfied this condition for the provision of Support Services.

(ii) Support Services will terminate unless renewed annually by payment of the then-current annual support charge and by successful completion of the Support Services Subscription Form again.

(iii) By completion of the Support Services Subscription Form you consent to the terms of the Kaspersky Lab Privacy Policy, which is attached to this Agreement, and you explicitly consent to the transfer of data to other countries outside your own as set out in the Privacy Policy.

(iv) "Support Services" means

(a) Daily updates of the anti-virus database;

(b) Free software updates, including version upgrades;

(c) Extended technical support via e-mail and phone hotline provided by Vendor and/or Reseller;

(d) Virus detection and disinfection updates 24 hours per day.

4. Ownership Rights. The Software is protected by copyright laws. Kaspersky Lab and its suppliers own and retain all rights, titles and interests in and to the Software, including all copyrights, patents, trademarks and other intellectual property rights therein. Your possession, installation, or use of the Software does not transfer any title to the intellectual property in the Software to you, and you will not acquire any rights to the Software except as expressly set forth in this Agreement.

5. Confidentiality. You agree that the Software and the Documentation, including the specific design and structure of individual programs and the Key Identification File, constitute confidential proprietary information of Kaspersky Lab. You shall not disclose, provide, or otherwise make available such confidential information in any form to any third party without the prior written consent of Kaspersky Lab. You shall implement reasonable security measures to protect such confidential information, but without limitation to the foregoing shall use best endeavours to maintain the security of the Key Identification File.

6. Limited Warranty

(i) Kaspersky Lab warrants that for 90 days from first download or installation the Software will perform substantially in accordance with the functionality described in the Documentation when operated properly and in the manner specified in the Documentation.

(ii) You accept all responsibility for the selection of this Software to meet your requirements. Kaspersky Lab does not warrant that the Software and/or the Documentation will be suitable for such requirements nor that any use will be uninterrupted or error free;

(iii) Kaspersky Lab does not warrant that this Software identifies all known viruses, nor that the Software will not occasionally erroneously report a virus in a title not infected by that virus;

(iv) Your sole remedy and the entire liability of Kaspersky Lab for breach of the warranty at paragraph (i) will be at Kaspersky Lab option, to repair, replace or refund of the Software if reported to Kaspersky Lab or its designee during the warranty period. You shall provide all information as may be reasonably necessary to assist the Supplier in resolving the defective item;

(v) The warranty in (i) shall not apply if you (a) make or cause to be made any modifications to this Software without the consent of Kaspersky Lab, (b) use the Software in a manner for which it was not intended or (c) use the Software other than as permitted under this Agreement;

(vi) The warranties and conditions stated in this Agreement are in lieu of all other conditions, warranties or other terms concerning the supply or purported supply of, failure to supply or delay in supplying the Software or the Documentation which might but for this paragraph (v) have effect between the Kaspersky Lab and you or would otherwise be implied into or incorporated into this Agreement or any collateral contract, whether by statute, common law or otherwise, all of which are hereby excluded (including, without limitation, the implied conditions, warranties or other terms as to satisfactory quality, fitness for purpose or as to the use of reasonable skill and care).

7. Liability

(i) Nothing in this Agreement shall exclude or limit Kaspersky Lab's liability for (i) the tort of deceit, (ii) death or personal injury caused by its breach of a common law duty of care or any negligent breach of a term of this Agreement, (iii) any breach of the obligations implied by s.12 Sale of Goods Act 1979 or s.2 Supply of Goods and Services Act 1982 or (iv) any liability which cannot be excluded by law.

(ii) Subject to paragraph (i), the Supplier shall bear no liability (whether in contract, tort, restitution or otherwise) for any of the following losses or damage (whether such losses or damage were foreseen, foreseeable, known or otherwise):

(a) Loss of revenue;

(b) Loss of actual or anticipated profits (including for loss of profits on contracts);

(c) Loss of the use of money;

(d) Loss of anticipated savings;

(e) Loss of business;

(f) Loss of opportunity;

(g) Loss of goodwill;

(h) Loss of reputation;

(i) Loss of, damage to or corruption of data, or:

(j) Any indirect or consequential loss or damage howsoever caused (including, for the avoidance of doubt, where such loss or damage is of the type specified in paragraph (ii), (a) to (ii), (i).

(iii) Subject to paragraph (i), the liability of Kaspersky Lab (whether in contract, tort, restitution or otherwise) arising out of or in connection with the supply of the Software shall in no circumstances exceed a sum equal to the amount equally paid by you for the Software.

8. The construction and interpretation of this Agreement shall be governed in accordance with the laws of England and Wales. The parties hereby submit to the jurisdiction of the courts of England and Wales save that Kaspersky Lab as claimant shall be entitled to initiate proceedings in any court of competent jurisdiction.

9. (i) This Agreement contains the entire understanding between the parties with respect to the subject matter hereof and supersedes all and any prior understandings, undertakings and promises between you and Kaspersky Lab, whether oral or in writing, which have been given or may be implied from anything written or said in negotiations between us or our representatives prior to this Agreement and all prior agreements between the parties relating to the matters aforesaid shall cease to have effect as from the Effective Date. Save as provided in paragraphs (ii) - (iii), you shall not have any remedy in respect of an untrue statement made to you upon which you relied in entering into this Agreement ("Misrepresentation") and Kaspersky Lab shall not have any liability to the other than pursuant to the express terms of this Agreement.

(ii) Nothing in this Agreement shall exclude or limit Kaspersky Lab's liability for any Misrepresentation made thereby if aware that it was untrue.

(iii) The liability of Kaspersky Lab for Misrepresentation as a fundamental matter, including a matter fundamental to the maker's ability to perform its obligations under this Agreement, shall be subject to the limitation of liability set out in paragraph 7(iii).