VirusScan for UNIX

VERSION 4.24.0
TRADEMARK ATTRIBUTIONS

Active Firewall, Active Security, Active Security (in Katakana), ActiveHelp, ActiveShield, AntiVirus Anyware and design, Bomb Shelter, Certified Network Expert, Clean-Up, CleanUp Wizard, CNX, CNX Certification Certified Network Expert and design, Design (stylized N), Disk Minder, Distributed Sniffer System, Distributed Sniffer System (in Katakana), Dr Solomon’s, Dr Solomon’s label, Enterprise SecureCast, Enterprise SecureCast (in Katakana), Event Orchestrator, EZ SetUp, First Aid, ForceField, GMT, GroupShield, GroupShield (in Katakana), Guard Dog, HelpDesk, HomeGuard, Hunter, LANGuru, LANGuru (in Katakana), M and design, Magic Solutions, Magic Solutions (in Katakana), Magic University, MagicSpy, MagicTree, McAfee, McAfee (in Katakana), McAfee and design, McAfee.com, MultiMedia Cloaking, Net Tools, Net Tools (in Katakana), NetCrypto, NetScan, NetShield, NetStalker, Network Associates, NetXray, NotesGuard, Nuts & Bolts, Oil Change, PC Medic, PCNotary, PrimeSupport, Recoverkey, Recoverkey – International, Registry Wizard, ReportMagic, Router PM, Safe & Sound, SalesMagic, SecureCast, Service Level Manager, ServiceMagic, SmartDesk, Sniffer, Sniffer (in Hangul), Stalker, SupportMagic, TIS, TMEG, Total Network Security, Total Network Visibility, Total Network Visibility (in Katakana), Total Service Desk, Total Virus Defense, Trusted Mail, UnInstaller, Virex, Virus Forum, VirusScan, VirusScan, WebScan, WebShield, WebShield (in Katakana), WebSniffer, WebStalker, WebWall, Who’s Watching Your Network, WinGauge, Your E-Business Defender, ZAC 2000, Zip Manager are registered trademarks of Network Associates, Inc. and/or its affiliates in the US and/or other countries. All other registered and unregistered trademarks in this document are the sole property of their respective owners.

This product includes or may include software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)

This product includes or may include cryptographic software written by Eric Young. (eay@cryptsoft.com)

LICENSE AGREEMENT

NOTICE TO ALL USERS: CAREFULLY READ THE APPROPRIATE LEGAL AGREEMENT CORRESPONDING TO THE LICENSE YOU PURCHASED, WHICH SETS FORTH THE GENERAL TERMS AND CONDITIONS FOR THE USE OF THE LICENSED SOFTWARE. IF YOU DO NOT KNOW WHICH TYPE OF LICENSE YOU HAVE ACQUIRED, PLEASE CONSULT THE SALES AND OTHER RELATED LICENSE GRANT OR PURCHASE ORDER DOCUMENTS THAT ACCOMPANY YOUR SOFTWARE PACKAGING OR THAT YOU HAVE RECEIVED SEPARATELY AS PART OF THE PURCHASE (AS A BOOKLET, A FILE ON THE PRODUCT CD, OR A FILE AVAILABLE ON THE WEB SITE FROM WHICH YOU DOWNLOADED THE SOFTWARE PACKAGE). IF YOU DO NOT AGREE TO ALL OF THE TERMS SET FORTH IN THE AGREEMENT, DO NOT INSTALL THE SOFTWARE. IF APPLICABLE, YOU MAY RETURN THE PRODUCT TO NETWORK ASSOCIATES OR THE PLACE OF PURCHASE FOR A FULL REFUND.
## Contents

**Preface** ................................................................. 5
- Audience ................................................................. 5
- Getting more information ........................................... 6
- Contacting McAfee and Network Associates .......................... 7

**Introduction** ......................................................... 9
- What’s new in this release ............................................... 10
- Configurable cache for file reading ................................ 10
- Support for plain-text mailboxes .................................... 10

**Installing VirusScan for UNIX** .................................... 11
- Before you begin ....................................................... 11
  - About the distributions ............................................. 11
  - Installation requirements .......................................... 12
  - Other recommendations ............................................. 12
- Installing ................................................................. 12
  - Troubleshooting during installation .............................. 14
- Testing your installation .............................................. 14
  - Troubleshooting when scanning ................................ 15
- Removing the program ................................................ 16

**Using VirusScan for UNIX** .......................................... 17
- Running an on-demand scan .......................................... 17
  - Command-line conventions ....................................... 18
  - General hints and tips ............................................. 18
- Configuring scans ..................................................... 19
- Scheduling scans ...................................................... 21
- Handling viruses ...................................................... 21
- Using heuristic analysis ............................................ 22
- Handling an infected file that cannot be cleaned ............... 23
- Producing reports .................................................... 24
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choosing the options</td>
<td>25</td>
</tr>
<tr>
<td>Scanning options</td>
<td>26</td>
</tr>
<tr>
<td>Response options</td>
<td>31</td>
</tr>
<tr>
<td>General options</td>
<td>32</td>
</tr>
<tr>
<td>Options in alphabetic order</td>
<td>33</td>
</tr>
<tr>
<td>Exit codes</td>
<td>36</td>
</tr>
<tr>
<td>Preventing Virus Infection</td>
<td>37</td>
</tr>
<tr>
<td>Detecting new and unidentified viruses</td>
<td>37</td>
</tr>
<tr>
<td>Why do I need new DAT files?</td>
<td>38</td>
</tr>
<tr>
<td>Updating your DAT files</td>
<td>38</td>
</tr>
<tr>
<td>Sample update script for UNIX</td>
<td>39</td>
</tr>
<tr>
<td>Sample update script for Perl</td>
<td>41</td>
</tr>
<tr>
<td>Index</td>
<td>45</td>
</tr>
</tbody>
</table>
This Product Guide introduces McAfee VirusScan for UNIX® software version 4.24.0, and provides the following information:

- Detailed instructions for installing the software.
- Descriptions of all new features in this release of the software.
- Descriptions of all product features.
- Detailed instructions for configuring and deploying the software.
- Procedures for performing tasks.

**Audience**

This information is intended primarily for two audiences:

- Network administrators who are responsible for the company’s anti-virus and security program.
- Users who are responsible for updating virus definition (DAT) files on their workstation, or configuring the software’s detection options.
Getting more information

**Help**  Product information in the Help system that is accessed from within the application on its “man” pages.

**Release Notes**  *README file.* Product information, system requirements, resolved issues, any known issues, and last-minute additions or changes to the product or its documentation.

Available as a .TXT file from either the product CD or the McAfee download site.

**Contact**  A list of phone numbers, street addresses, web addresses, and fax numbers for Network Associates offices in the United States and around the world. Also provides contact information for services and resources, including:

- Technical Support
- Customer Service
- Download Support
- AVERT Anti-Virus Research Site
- McAfee Beta Site
- On-Site Training
- Network Associates Offices Worldwide
Contacting McAfee and Network Associates

Technical Support http://knowledge.nai.com

McAfee Beta Site www.mcafeeb2b.com/beta/

AVERT Anti-Virus Emergency Response Team www.mcafeeb2b.com/naicommon/avert/default.asp

Download Site www.mcafeeb2b.com/naicommon/download/
DAT File Updates www.mcafeeb2b.com/naicommon/download/dats/find.asp
Product Upgrades www.mcafeeb2b.com/naicommon/download/upgrade/login.asp
Valid grant number required.
Contact Network Associates Customer Service.

On-Site Training www.mcafeeb2b.com/services/mcafee-training/default.asp

Network Associates Customer Service:
E-mail services_corporate_division@nai.com
Web www.nai.com
www.mcafeeb2b.com

US, Canada, and Latin America toll-free:
Phone +1-888-VIRUS NO or +1-888-847-8766
Monday – Friday, 8 a.m. – 8 p.m., Central Time

For additional information on contacting Network Associates and McAfee — including toll-free numbers for other geographic areas — see the Contact file that accompanies this product release.
VirusScan for UNIX detects and removes viruses on UNIX-based systems. The scanner runs from a command-line prompt, and provides an alternative to scanners that use a graphical user interface (GUI). Both types of scanner use the same anti-virus software.

The scanner acts as an interface to the powerful anti-virus scanning engine — the engine common to all our anti-virus products.

The UNIX operating system is a secure environment, relatively unaffected by computer viruses. However, DOS and Windows environments are less secure and more susceptible to virus infections. Those viruses do not affect UNIX systems, but many DOS-based and Windows-based computers are connected to UNIX servers. Although a server itself is not affected, it can pass viruses on to its clients. Rather than trying to block viruses at each computer connected to a UNIX server, you can install the VirusScan for UNIX software and use it as an efficient centralized solution.

The command-line scanner enables you to search for viruses in any folder or file in your computer “on demand” — in other words, at any time. The command-line scanner also features options that can alert you when they detect a virus or take a variety of automatic actions.

When kept up-to-date with the latest virus-definition (DAT) files, the scanner is an important part of your network security. We recommend that you set up an anti-virus security policy for your network, incorporating as many protective measures as possible.
What's new in this release

This release of VirusScan for UNIX includes the following new features:

- Configurable cache for file reading.
- Support for plain-text mailboxes.

Configurable cache for file reading

Previous release When making file reads, the engine normally allocates a small amount of memory (or 'cache') as determined by the operating system.

Current release A larger amount of cache can now be specified.

Benefits Files (especially large archives) can be scanned faster.

For more information See the description of the new option, --afc on page 26.

Support for plain-text mailboxes

Previous release Plain-text mailboxes were not scanned.

Current release Scanning of plain text mailboxes is now supported.

Benefits Scanning is now available for Eudora, PINE, and Netscape.

Where to find This feature is enabled by the option, --mailbox.

For more information See --mailbox on page 28.
Before you begin

We distribute the VirusScan for UNIX software in two ways — on a CD, and as an archived file that you can download from the Network Associates web site or from other electronic services.

After you have downloaded a file or placed your disk in your CD drive, the installation steps are the same for each type of distribution version.

Review the Installation requirements on page 12 to verify that the software will run on your system, then follow the installation steps.

About the distributions

VirusScan for UNIX software comes in several distribution versions, one for each supported operating system.

- Sun Microsystems Solaris for SPARC architecture, versions 2.5.1, 2.6, 7 and 8, with all recommended patches installed.
- Hewlett-Packard HP-UX 10.20, 10.30, 11.x, and HP-UX 11i, with all recommended patches installed.
- IBM AIX version 4.2.1, 4.3.x, and AIX 5.0L, with all recommended patches installed.
- Linux version 2.x kernels on Intel-based systems, Linux version 2.0, 2.2 and 2.4 kernels (with libc6) and the stdc++ library version 2.8.
- Santa Cruz Operation (SCO) OpenServer Release 5 and SCO UnixWare 7.1.1. The SCO UnixWare Binary Compatibility Module (BCM) must be installed.
- FreeBSD version 3.2 and 4.3.

If you install VirusScan for UNIX software from CD, each version is in its own directory. Each distribution has its own installation script.
Installing VirusScan for UNIX

Installation requirements

To install and run the software, you need the following:

- The correct version of the UNIX distribution that you require, installed and running correctly on the target machine. See About the distributions on page 11 for information.
- 4MB of free hard disk space for a full installation.
- A CD drive, if you are not downloading the software from a web site.

Other recommendations

- To install the software and perform on-demand scan operations of your file system, we recommend that you have root account permissions.
- To take full advantage of the regular updates to DAT files from our web site, you need an Internet connection, either through your local area network, or via a high-speed modem and an Internet Service Provider.

Installing

This example shows how to install the software on the Solaris distribution. To install other distributions, substitute the correct filename (for example vsun4180.tar.Z) where the example specifies the distribution file.

To start the installation script:

1. Download the appropriate VirusScan for UNIX software distribution from the Network Associates web site or insert the McAfee installation CD.

   If you are using the McAfee installation CD to obtain the software, you can mount the CD on to the file system.

2. Copy the distribution file to a directory on your system.

   NOTE
   We recommend that you use a separate (possibly a temporary) directory — not the directory where you intend to install the software.

3. Enter this line at the command prompt to decompress the file:

   
   zcat <distribution file> | tar -xf -
4 Enter this line at the command prompt to execute the installation script:

`./install-uvscan installation-directory`

Here, the `installation-directory` is the directory where you want to install the software.

If you do not specify an installation directory, the software is installed in `/usr/local/uvscan`.

If the installation directory does not exist, the installation script prompts you to create it. If you do not create the installation directory, the installation cannot continue.

5 The installation script asks whether you want to place links to the executable, the shared library and the man page. Enter Y to create each link, or N to skip the step.

We recommend that you create these links. Otherwise, you will need to set one of the following environment variables to contain the installation directory:

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM AIX</td>
<td>LIBPATH</td>
</tr>
<tr>
<td>FreeBSD</td>
<td>LD_LIBRARY_PATH</td>
</tr>
<tr>
<td>HP-UX</td>
<td>SHLIB_PATH</td>
</tr>
<tr>
<td>Linux</td>
<td>LD_LIBRARY_PATH</td>
</tr>
<tr>
<td>SCO OpenServer</td>
<td>LD_LIBRARY_PATH</td>
</tr>
<tr>
<td>Sun Solaris</td>
<td>LD_LIBRARY_PATH</td>
</tr>
</tbody>
</table>

**NOTE**

The program also looks in the `/usr/lib` or `/lib` directory or the current directory for the shared library.

6 The installation program copies the program files to your hard disk, then scans your home directory.

If the software discovers a virus, see *Handling viruses on page 21* to learn about the actions you can take.

If the installation fails, see *Troubleshooting during installation on page 14* to learn about possible errors and suggested courses of action.
Troubleshooting during installation

The following table lists the most common error messages returned if the installation fails. The table also suggests a likely reason for the error and recommends any solutions.

<table>
<thead>
<tr>
<th>Error</th>
<th>Cause or action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failed to create install_dir</td>
<td>Verify that you have permission to create the installation directory.</td>
</tr>
<tr>
<td>Cannot write to install_dir</td>
<td>Verify that you have permission to create installation directory.</td>
</tr>
<tr>
<td>The install_dir exists, but is not a subdirectory</td>
<td>Choose another installation directory.</td>
</tr>
<tr>
<td>&lt;file&gt; is missing</td>
<td>The file might not exist.</td>
</tr>
<tr>
<td>&lt;file&gt; is not correct</td>
<td>The file did not install correctly.</td>
</tr>
</tbody>
</table>

Testing your installation

After it is installed, the program is ready to scan your system for infected files. You can run a test to determine that the program is installed correctly and can scan properly for viruses. The test was developed by EICAR, a coalition of anti-virus vendors headquartered in Europe, as a method for testing any anti-virus software installation.

To test your installation:

1. Open a standard text editor, then type the following line:
   
   `X5O!P%@AP\(4\(PZX54(P^)(7CC)7)\$EICAR-STD-ANTIVIRUS-TEST-FILE!$H+H*`

   **NOTE**
   The line must appear as one line in the window of your text editor.

2. Save the file with the name EICAR.COM. The file size will be 68 or 70 bytes.

3. Enter the following command to scan the EICAR.COM file:
   
   `uvscan -v eicar.com`
When the program examines this file, it reports finding the EICAR test file, but you will not be able to clean or repair it.

**NOTE**
The EICAR test file *does not contain a virus*—it cannot spread or infect other files, or otherwise harm your system.

4 When you have finished testing your installation, delete the test file to avoid alarming other users.

If the software appears not to be working correctly, check that you have Read permissions on the test file.

**Troubleshooting when scanning**
The following table lists the most common error messages returned if the `uvscan` program fails when scanning. The table also suggests a likely reason for the error and recommends solutions.

<table>
<thead>
<tr>
<th>Program message</th>
<th>Remedy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cannot find shared object</td>
<td>• AIX — Install the correct version of -xlC.rte. The program does not run on versions before 4.0</td>
</tr>
<tr>
<td></td>
<td>• HP-UX — Install the aCC run-time patch.</td>
</tr>
<tr>
<td></td>
<td>• Linux — Install LibC6; LibC5 is not supported.</td>
</tr>
<tr>
<td>Unable to find shared library</td>
<td>Set the appropriate environment variable:</td>
</tr>
<tr>
<td></td>
<td>• For AIX, use LIBPATH.</td>
</tr>
<tr>
<td></td>
<td>• For HP-UX, use SHLIB_PATH.</td>
</tr>
<tr>
<td></td>
<td>• For SCO OpenServer, Solaris, FreeBSD and Linux, use LD_LIBRARY_PATH.</td>
</tr>
<tr>
<td>Cannot execute: permission denied</td>
<td>Check the file permissions. Incorrect file permissions can prevent the program running correctly. All executables (including the shared libraries) must have Read and Execute permissions (r_x), but we recommend rwxr_xr_x</td>
</tr>
<tr>
<td></td>
<td>All DAT files must have read permissions.</td>
</tr>
<tr>
<td>Missing or invalid DAT files</td>
<td>Re-install the DAT files.</td>
</tr>
<tr>
<td>The program has been altered; please replace with a good copy</td>
<td>Re-install from the original media; the program might be infected.</td>
</tr>
</tbody>
</table>
Removing the program

A script is installed at the same time as the VirusScan for UNIX software, which enables you to remove the product quickly and easily.

To remove the product from your system:

1. Run the script `uninstall-uvscan`, which is in the VirusScan for UNIX program directory. For example, enter the following command at the command prompt:

   `/usr/local/uvscan/uninstall-uvscan`

2. Delete the script `uninstall-uvscan` from the program directory to remove the program completely from your system.

If you created your own links to the program and a shared library path when you installed the software, you must remove those links yourself.

If you are an administrator, ensure that your users cannot accidentally remove their VirusScan for UNIX software.

Removing the software leaves your computer unprotected against virus attack. Remove the product only when you are sure that you can upgrade quickly to a new version.
VirusScan for UNIX provides virus scanning from a command line. This chapter describes how to use its features and customize the program to meet your needs.

The following features offer optimum protection for your computer and network:

- On-demand scanning options let you start a scan immediately or schedule automatic scans.
- Advanced heuristic analysis detects previously unknown macro viruses and program viruses.
- Updates to virus-definition files and upgrades to program components ensure that the program has the most current scanning technology to deal with viruses as they emerge.

Later sections in this guide describe each of these features in detail.

**Running an on-demand scan**

You can scan any file or directory on your file system from the command line by adding options to the basic command.

Only the Intel-based FreeBSD, SCO-UNIX and Linux distributions of the VirusScan for UNIX program can scan for boot-sector viruses.

When run without options, the program simply displays a brief summary of its options. When run with only a directory name specified, the program scans every file in that directory only, and issues a message if any infected file is found. The options fall into these main groups:

- **Scanning options** — determine how and where the scanner looks for infected files. See page 26.
- **Response options** — determine how the scanner responds to any infected files. See page 31.
- **General options** — determine how the scanner reports its scanning activities. See page 32.

Each group of options appears in its own table with a description of its function. See **Choosing the options on page 25** for details.
**Command-line conventions**

Use these conventions to add options to the command line:

- Type each option in lower case and separate each with spaces.
- Do not use any option more than once on the command line.
- Follow the syntax correctly. The UNIX operating system is case-sensitive.
- Type single consecutive options as one option. For example, instead of typing this:
  
  `-c -r --one-file-system`

  you can type this:

  `-cr --one-file-system`

- To start the program, at the command prompt, enter:

  `uvscan`

- To have the program examine a specific file or list of files, add the target directories or files to the command line after `uvscan`. You can also create a text file that lists your target files, then add the name of the text file to the command line. See *Configuring scans* on page 19.

  By default, the program examines all files, no matter what their extensions. You can limit your scan by adding only those extensions you want to examine to the command line after the `--extensions` option, or you may exclude certain files from scans with the `--exclude` option. See *Choosing the options* on page 25 for details.

**General hints and tips**

- To display a list of all the options, each with a short description of their features, enter the command:

  `uvscan --help`

- To display a list of all the viruses that the program detects, enter the command:

  `uvscan --virus-list`

- To display information about the version of the program, enter the command:

  `uvscan --version`
To scan all folders within a folder with maximum security, enter the command:

```
uvscan -r --secure /usr/myself/myfolder
```

To ensure maximum protection from virus attack, you must regularly update your DAT files. See Preventing Virus Infection on page 37 for details.

## Configuring scans

Instead of running each scan with all its options directly from the command line, you can configure a scan with the options you choose, then save it in a text file as a scan task.

This allows you to run complete scans with ease, and at any time. Your scan task specifies the actions that are performed when a virus is detected.

### To configure a scan:

1. Choose the command options that you want to use.
   
   See Choosing the options on page 25 for a description of available options.

2. Type the command options into a text editor just as you might on the command line.

3. Save the text as a file.

4. Enter one of these lines at the command prompt:

   ```
   uvscan --load <file> <target>
   uvscan --config <file> <target>
   ```

   Here, `<file>` is the name of the text file you created, and `<target>` is the file or directory you want to scan.

   If the scanner detects no virus infection, it displays no output.

To learn how to specify the options, see Command-line conventions on page 18.
Using VirusScan for UNIX

Example 1
To scan files in the /usr/dos directory according to the settings you stored in the task file, /usr/local/config1, enter the command:

```
uvscan --load /usr/local/config1 /usr/dos
```

The contents of the task file /usr/local/config1, are:

```
-m /viruses --ignore-compressed --maxfilesize 4
```

They instruct the scan to move any infected files to /viruses, to ignore any compressed files in the target directory, and to examine only files smaller than 4MB.

As an alternative, you can arrange the contents of the task file as single lines:

```
-m /usr/local/viruses
--ignore-compressed
--maxfilesize 4
```

Example 2
To scan only files smaller than 4MB and to ignore any compressed files in three separate directories, enter the command:

```
uvscan --load /usr/local/config1 --file mylist
```

The contents of the task file /usr/local/config1, are:

```
--ignore-compressed
--maxfilesize 4
```

The contents of the other file, mylist, are:

```
/usr/local/bin
/temp
/etc
```
Scheduling scans

You can use the UNIX cron scheduler to run automated scans. Cron stores the scheduling commands in its crontab files. For further information about cron and crontab, refer to your UNIX documentation or view the help text, using these commands: man cron and man crontab.

Examples

To schedule a scan to run at 18:30 every weekday, add this line to your crontab file:

```
30 18 * * 1-5 /usr/local/bin/uvscan
```

To schedule a scan to run and produce a summary at 11:50 p.m. every Sunday, add this line to your crontab file:

```
50 23 * * 0 /usr/local/bin/uvscan --summary
```

To schedule a scan to run on the uz directory at 10:15 a.m. every Saturday in accordance with options specified in a configuration file conf1, add this line to your crontab file:

```
15 10 * * 6 /usr/local/bin/uvscan --load conf1 /uz
```

To schedule a scan to run at 8:45 a.m. every Monday on the files specified in the file mylist, add this line to your crontab file:

```
45 8 * * 1 /usr/local/bin/uvscan --f /usr/local/mylist
```

Handling viruses

If the scanner discovers a virus while scanning, it returns exit code number 13. See Exit codes on page 36 for a full description of each code.

To clean infected files or directories, or move them to a quarantine location on your network, you can configure your scans using one or more response options, which are described in Table 3-3 on page 31.

The following examples show how you can use these options to respond to a virus attack. The examples assume that the scanner is available in your search path.

Example 1

To scan and clean all files in the /usr/dos directory and all of its subdirectories, enter the command:

```
uvscan -cr /usr/dos
```

The VirusScan for UNIX program (uvscan.exe) scans /usr/dos and its subdirectories automatically, and cleans any infected files that it encounters.
Example 2
To scan and clean all files in the /usr/dos directory and its subdirectories, but ignore any other file systems that are mounted, enter the command:

```
uvscan -cr --one-file-system /usr/dos
```

Example 3
To scan all files, except compressed files, in the /usr/dos directory and its subdirectories, and to move any infected files to /viruses, enter the command:

```
uvscan -m /viruses -r --ignore-compressed /usr/dos
```

Example 4
To scan a file with a name prefixed with “-”, enter the command:

```
uvscan -c -v -myfile
```

The program scans the named file. It cleans any detected viruses and issues a progress message. This format avoids confusion between the names of the options and the name of the target. Without the “-” option, the uvscan command appears to have three options and no target:

```
uvscan -c -v -myfile
```

Using heuristic analysis

An anti-virus scanner uses two techniques to detect viruses: signature matching and heuristic analysis.

A virus signature is simply a binary pattern that is found in a virus-infected file. Using information in the DAT files, the scanner searches for those patterns.

However, this approach cannot detect a new virus because its signature is not yet known, therefore the scanner use another technique — heuristic analysis.

Programs, documents or e-mail messages that carry a virus often have distinctive features. They might attempt unprompted modification of files, invoke mail clients, or use other means to replicate themselves. The scanner analyzes the program code to detect these kinds of computer instructions. The scanner also searches for “legitimate,” non-virus-like behavior, such as prompting the user before taking action, and thereby avoids raising false alarms.

In an attempt to avoid being detected, some viruses are encrypted. Each computer instruction is simply a binary number, but the computer does not use all the possible numbers. By searching for unexpected numbers inside a program file, the scanner can detect an encrypted virus. By using these two techniques, the scanner can detect both known viruses and many new viruses and variants. Options that use heuristic analysis include ---analyze, --manalyze, --panalyze.
Handling an infected file that cannot be cleaned

If the scanner cannot clean an infected file, it renames the file to prevent its use. When a file is renamed, only the file extension (typically three letters) is changed. The following table shows the method of renaming.

<table>
<thead>
<tr>
<th>Original</th>
<th>Renamed</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not v??</td>
<td>v??</td>
<td>File extensions that do not start with v are renamed with v as the initial letter of the file extension. For example, myfile.doc becomes myfile.voc.</td>
</tr>
<tr>
<td>v??</td>
<td>vir</td>
<td>File extensions that start with v are renamed as .vir. For example, myfile.vbs becomes myfile.vir.</td>
</tr>
<tr>
<td>vir, v01-v99</td>
<td></td>
<td>These files are recognized as already infected, and are not renamed again.</td>
</tr>
<tr>
<td>&lt;blank&gt;</td>
<td>vir</td>
<td>Files with no extensions are given the extension, .vir.</td>
</tr>
</tbody>
</table>

For example, if an infected file called bad.com is found, the scanner attempts to rename the file to bad.vom. However, if a file of that name already exists in the directory, the scanner attempts to rename the file to bad.vir, bad.v01, or bad.v02, and so on.

For file extensions with more than three letters, the name is usually not truncated. For example, notepad.class becomes notepad.vlass. However, an infected file called water.vapor becomes water.vir.
Producing reports

The program might take some time to complete a scan, particularly over many directories and files. However, the scanner can keep you informed of its progress, any viruses it finds, and its response to them.

The program displays this information on your screen if you add the --summary or --verbose options to the command line. To learn more about each option, see Response options on page 31.

The --verbose option tells you which files the program is examining.

When the scan finishes, the --summary option identifies the following:

- How many files were scanned.
- How many files were cleaned.
- How many files were not scanned.
- How many infected files were found.

Example

In the report information below, both the --summary and --verbose options were used when scanning files in the /usr/data directory.

```
$ uvscan --summary --verbose /usr/data

Scanning /usr/data/*
Scanning file /usr/data/command.com
Scanning file /usr/data/grep.com
Summary report on /usr/data/*
File(s)
  Total files: ...........       2
  Clean: .................       2
  Not scanned: ...........       0
  Possibly Infected: .....       0
```
Choosing the options

The following sections describe the options you can use to target your scan:

- *General options* on page 32.
- *Options in alphabetic order* on page 33.

The descriptions use these conventions to identify the options or required parameters:

- Short versions of each command option appear after a single dash (-).
- Long versions of each command option, if any, appear after two dashes (--).
- Variables, such as filenames or paths, appear in italics within brackets <>.

To learn how to add these options to the command line, see *Command-line conventions* on page 18.
Using VirusScan for UNIX

Scanning options

Scanning options describe how and where each scan looks for infected files. You can use a combination of these options to customize the scan to suit your needs.

Table 3-2. Scanning options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--afc &lt;size&gt;</td>
<td>Specify the maximum cache size for archive files. When handling compressed files, the program first de-compresses them into memory before scanning. By default, the cache sizes are 24Mb for client computers and 64Mb for servers. A larger cache size enables compressed files to be scanned quickly, but can slow the performance of a computer that has low memory. The range of sizes allowed is 8Mb to 512Mb. Specify the size in megabytes. For example, --afc 64 specifies up to 64Mb of cache.</td>
</tr>
<tr>
<td>--allele</td>
<td>Check every file for OLE objects.</td>
</tr>
<tr>
<td>--analyze</td>
<td>Use heuristic analysis to find possible new viruses in “clean” files. This step occurs after the program has checked the file for other viruses.</td>
</tr>
<tr>
<td>--atime-preserve</td>
<td>Preserve the last-accessed time and date for files that are scanned. Some backup software archives only changed files, and determines this information from each file’s last-accessed date (or ‘a-time’). Normally, scanning changes that date. This option will preserve the date, enabling the backup software to work as intended. Sometimes when this option is used, the file date is not preserved; if a file contains a virus, or the scan was started by a user who does not own the file, the file date is changed.</td>
</tr>
<tr>
<td>--config &lt;file&gt;</td>
<td>Run the options specified in &lt;file&gt;. You cannot nest configuration files within other configuration files.</td>
</tr>
<tr>
<td>--data-directory</td>
<td>Specify the location of the DAT files — scan.dat, names.dat, and clean.dat. If you do not use this option in the command line, the program looks in the same directory from where it was executed. If it cannot find these data files, the program issues exit code 6.</td>
</tr>
</tbody>
</table>
Examine files that have the specified extension. You can specify as many extensions as you want. Separate each with a comma, but without a space. If you choose this option, the program scans only susceptible files, files with execute permissions and those you specify here. To see the list of susceptible files, use the **--extlist** option.

--extra <file>

Specify the location of any extra.dat file. If you do not use this option in the command line, the program looks in the same directory from where it was executed. If it cannot find this file, the program issues exit code 6.

--fam

Locate all files that have macros. Use this option with caution if you use it with the **--cleandocall** or **--dam** options.

--file <file>  
-f <file>

Scan the directories or files as specified in `<file>`.

--floppy

Scan the boot sector of the diskette (floppy disk) in drive A or B. This option is for Intel-based UNIX systems only, namely FreeBSD, SCO-UNIX and Linux.

--ignore-compressed  
--nocomp

Ignore compressed files. By default, the program scans files saved in these compression formats: ICE, LZEXE, PKLITE, Cryptcom, COM2EXE, Diet, Teledisk, Microsoft Expand and GZIP. This option reduces the scanning time but increases the virus threat because many file types are not scanned. By default, the program scans compressed files.
Using VirusScan for UNIX

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--ignore-links</td>
<td>Do not resolve any symbolic links and do not scan the link targets. Normally, the program follows each symbolic link and scans the linked file.</td>
</tr>
<tr>
<td>--load &lt;file&gt;</td>
<td>See --config option.</td>
</tr>
<tr>
<td>--mailbox</td>
<td>Scan plain-text mailboxes. These include Eudora, PINE, and Netscape. Most mailboxes will be in MIME format, and therefore the --mime option is also required.</td>
</tr>
<tr>
<td>--manalyze</td>
<td>Use heuristic analysis to identify potential macro viruses. This is a subset of --analyze. See Using heuristic analysis on page 22 for more information.</td>
</tr>
<tr>
<td>--macro-heuristics</td>
<td>This is a subset of --analyze. See Using heuristic analysis on page 22 for more information.</td>
</tr>
<tr>
<td>--maxfilesize &lt;size&gt;</td>
<td>Examine only those files smaller than the specified size. Specify the file size in megabytes. For example, maxfilesize 5 means scan only files that are smaller than 5 MB.</td>
</tr>
<tr>
<td>--mime</td>
<td>Scan MIME-encoded files. This type of file is not scanned by default.</td>
</tr>
<tr>
<td>--noboot</td>
<td>Do not scan the boot sector.</td>
</tr>
<tr>
<td>--nodecrypt</td>
<td>Do not decrypt Microsoft Office compound documents that are password-protected. By default, macros inside password-protected compound documents are scanned by employing “password cracking” techniques. If, for reasons of security, you do not require these techniques, use this option. Password cracking does not render the file readable.</td>
</tr>
<tr>
<td>--noexpire</td>
<td>Do not issue a warning if the DAT files are out of date.</td>
</tr>
<tr>
<td>--nodoc</td>
<td>Do not scan Microsoft Office document files.</td>
</tr>
<tr>
<td>--norename</td>
<td>Do not rename an infected file that cannot be repaired. See Handling an infected file that cannot be cleaned on page 23 for information about renaming.</td>
</tr>
</tbody>
</table>
**Using VirusScan for UNIX**

**Table 3-2. Scanning options (Continued)**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>--noscript</strong></td>
<td>Do not scan files that contain HTML, Javascript, Visual Basic, or Script Component Type Libraries. This type of file is normally scanned by default. Stand-alone Javascript and Visual Basic Script files will still be scanned.</td>
</tr>
<tr>
<td><strong>--one-file-system</strong></td>
<td>Scan an entire directory tree without scanning mounted file systems, if you use this option in conjunction with the <strong>--sub</strong> option. Normally, the program treats a mount point as a subdirectory and scans that file system. This option prevents the scan from running in subdirectories that are on a different file system to the original directory.</td>
</tr>
<tr>
<td><strong>--panalyze</strong></td>
<td>Use heuristic analysis to identify potential program viruses. By default, the program scans only for known viruses. The <strong>--panalyze</strong> option is a subset of <strong>--analyze</strong>, see Using heuristic analysis on page 22 for more information.</td>
</tr>
<tr>
<td><strong>--program</strong></td>
<td>Scan for malicious applications. Some widely available applications, such as “password crackers” can be used maliciously or can pose a security threat.</td>
</tr>
<tr>
<td><strong>-r</strong></td>
<td>Examine any subdirectories in addition to the specified target directory</td>
</tr>
<tr>
<td><strong>--recursive</strong></td>
<td></td>
</tr>
<tr>
<td><strong>--sub</strong></td>
<td></td>
</tr>
<tr>
<td><strong>--secure</strong></td>
<td>Examine all files, unzip archive files and use heuristic analysis. This option activates the <strong>--analyze</strong> and <strong>--unzip</strong> options. If the <strong>--selected</strong> and <strong>--extensions</strong> options are in the command line, they are ignored.</td>
</tr>
</tbody>
</table>
Using VirusScan for UNIX

Table 3-2. Scanning options (Continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| --selected  
 `-s`  | Look for viruses in any file that has execute permissions, and all files that are susceptible to virus infection.  
By default, all files are scanned. By scanning only files that are susceptible to virus infection, the program can scan a directory faster.  
To see the list of susceptible files, use the --extlist option. |
| --unzip  | Scan inside archive files, such as those saved in ZIP, LHA, PKarc, ARJ, TAR, CHM, and RAR formats.  
If used with --clean, this option attempts to clean non-compressed files inside ZIP files only. No other archive formats can be cleaned.  
The --clean option does not delete or rename infected files within ZIP files. It does not rename the ZIP file itself.  
The program cannot cleaned infected files found within any other archive format; you must first extract them from the archive file. |
Response options

These options determine how your scan responds to a virus infection. You can use a combination of these options to customize the scan. None of the options in this table occur automatically. To activate each option, specify it in the command line.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--clean</td>
<td>Automatically remove any viruses from infected files. By default, the program states only that infections were found but does not try to clean the infected file. If the program cannot clean the file, it displays a warning message. If you use this option, repeat the scan to ensure that there are no more infections.</td>
</tr>
<tr>
<td>-c</td>
<td></td>
</tr>
<tr>
<td>--cleandocall</td>
<td>Delete all macros in a file if an infected macro is found. If you suspect that a file is infected, you may remove all macros from the file to prevent any exposure to a virus. Use this option with caution when you also use the --fam option.</td>
</tr>
<tr>
<td>--dam</td>
<td></td>
</tr>
<tr>
<td>--delete</td>
<td>Automatically delete any infected files that are found.</td>
</tr>
<tr>
<td>--move &lt;directory&gt; -m &lt;directory&gt;</td>
<td>Move any infected files to a quarantine location as specified. When the program moves an infected file, it replicates the full directory path for the infected file inside the quarantine directory so you can determine the infected file's original location. If you use this option with --clean, the program copies the infected files to a quarantine location and tries to clean the original. If the program cannot clean the original, it deletes the file.</td>
</tr>
</tbody>
</table>
General options

These options provide help or give extra information about the scan. You may use a combination of these options to customize the scan. None of the options in this table occur automatically. To activate each option, specify it as part of the command line.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>Denote the end of the options and the start of the target to be scanned.</td>
</tr>
<tr>
<td>--extlist</td>
<td>Display a list of all file extensions which are susceptible to virus infection.</td>
</tr>
<tr>
<td></td>
<td>In other words, those file extensions that are scanned when --selected is set.</td>
</tr>
<tr>
<td>--help</td>
<td>List the most commonly used options, with a short description.</td>
</tr>
<tr>
<td>--summary</td>
<td>Produce a summary of the scan.</td>
</tr>
<tr>
<td></td>
<td>This includes the following:</td>
</tr>
<tr>
<td></td>
<td>• How many files were examined.</td>
</tr>
<tr>
<td></td>
<td>• How many infected files were found.</td>
</tr>
<tr>
<td></td>
<td>• How many viruses were removed from infected files.</td>
</tr>
<tr>
<td>--verbose</td>
<td>Display a progress summary during the scan.</td>
</tr>
<tr>
<td>--version</td>
<td>Display the program’s version number.</td>
</tr>
<tr>
<td>--virus-list</td>
<td>Display a list of all the viruses that the program can detect.</td>
</tr>
</tbody>
</table>
## Options in alphabetic order

For convenience, the options are repeated in this section in alphabetic order. For fuller descriptions, see the previous sections.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>See ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>Denote the end of the options and the start of the target to be scanned.</td>
<td>page 32</td>
</tr>
<tr>
<td>--afc &lt;size&gt;</td>
<td>Specify the maximum cache size for archive files.</td>
<td>page 26</td>
</tr>
<tr>
<td>--allole</td>
<td>Check every file for OLE objects.</td>
<td>page 26</td>
</tr>
<tr>
<td>--analyze</td>
<td>Same as --analyze.</td>
<td>page 26</td>
</tr>
<tr>
<td>--analyse</td>
<td>Use heuristic analysis to find possible new viruses in “clean” files.</td>
<td>page 26</td>
</tr>
<tr>
<td>--atime-preserve</td>
<td>Preserve the last-accessed time and date for files that are scanned.</td>
<td>page 26</td>
</tr>
<tr>
<td>-c</td>
<td>Same as --clean.</td>
<td>page 31</td>
</tr>
<tr>
<td>--clean</td>
<td>Automatically remove any viruses from infected files.</td>
<td>page 31</td>
</tr>
<tr>
<td>--cleandocall</td>
<td>Same as --dam.</td>
<td>page 31</td>
</tr>
<tr>
<td>--config &lt;file&gt;</td>
<td>Run the options specified in &lt;file&gt;.</td>
<td>page 26</td>
</tr>
<tr>
<td>-d &lt;directory&gt;</td>
<td>Same as --dat &lt;directory&gt;.</td>
<td>page 26</td>
</tr>
<tr>
<td>--dam</td>
<td>Delete all macros in a file if an infected macro is found.</td>
<td>page 31</td>
</tr>
<tr>
<td>--dat &lt;directory&gt;</td>
<td>Specify the location of the DAT files — scan.dat, names.dat, and clean.dat.</td>
<td>page 26</td>
</tr>
<tr>
<td>--data-directory</td>
<td>Same as --dat &lt;directory&gt;.</td>
<td>page 26</td>
</tr>
<tr>
<td>--delete</td>
<td>Automatically delete any infected files that are found.</td>
<td>page 31</td>
</tr>
<tr>
<td>-e</td>
<td>Same as --exit-on-error.</td>
<td>page 27</td>
</tr>
<tr>
<td>--exclude &lt;file&gt;</td>
<td>Exclude the directories or files from the scan as specified in &lt;file&gt;.</td>
<td>page 27</td>
</tr>
<tr>
<td>--exit-on-error</td>
<td>Quit and display an error message if an error is found.</td>
<td>page 27</td>
</tr>
<tr>
<td>--extensions &lt;EXT1[,EXT2,...]&gt;</td>
<td>Examine files that have the specified extension.</td>
<td>page 27</td>
</tr>
</tbody>
</table>
Table 3-5. Options in alphabetic order (Continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>See ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>--extlist</td>
<td>Display a list of all file extensions which are susceptible to virus infection.</td>
<td>page 32</td>
</tr>
<tr>
<td>--extra &lt;file&gt;</td>
<td>Specify the location of any extra.dat file.</td>
<td>page 27</td>
</tr>
<tr>
<td>-f &lt;file&gt;</td>
<td>Same as --file &lt;file&gt;.</td>
<td>page 27</td>
</tr>
<tr>
<td>--fam</td>
<td>Locate all files that have macros.</td>
<td>page 27</td>
</tr>
<tr>
<td>--file &lt;file&gt;</td>
<td>Scan the directories or files as specified in &lt;file&gt;.</td>
<td>page 27</td>
</tr>
<tr>
<td>--floppya</td>
<td>Scan the boot sector of the diskette (floppy disk) in drive A or B.</td>
<td>page 27</td>
</tr>
<tr>
<td>--floppyb</td>
<td>Same as --help.</td>
<td>page 32</td>
</tr>
<tr>
<td>-h</td>
<td>List the most commonly used options, with a short description.</td>
<td>page 32</td>
</tr>
<tr>
<td>--ignore-compressed</td>
<td>Ignore compressed files.</td>
<td>page 27</td>
</tr>
<tr>
<td>--ignore-links</td>
<td>Do not resolve any symbolic links and do not scan the link targets.</td>
<td>page 28</td>
</tr>
<tr>
<td>--load &lt;file&gt;</td>
<td>Same as --config &lt;file&gt;.</td>
<td>page 26</td>
</tr>
<tr>
<td>-m &lt;directory&gt;</td>
<td>Same as --move &lt;directory&gt;.</td>
<td>page 31</td>
</tr>
<tr>
<td>--macro-heuristics</td>
<td>Same as --manalyze.</td>
<td>page 28</td>
</tr>
<tr>
<td>--mailbox</td>
<td>Scan plain-text mailboxes.</td>
<td>page 28</td>
</tr>
<tr>
<td>--manalyse</td>
<td>Same as --manalyze.</td>
<td>page 28</td>
</tr>
<tr>
<td>--manalyze</td>
<td>Use heuristic analysis to identify potential macro viruses.</td>
<td>page 28</td>
</tr>
<tr>
<td>--maxfilesize &lt;size&gt;</td>
<td>Examine only those files smaller than the specified size.</td>
<td>page 28</td>
</tr>
<tr>
<td>--mime</td>
<td>Scan MIME-encoded files.</td>
<td>page 28</td>
</tr>
<tr>
<td>--move &lt;directory&gt;</td>
<td>Move any infected files to a quarantine location as specified.</td>
<td>page 31</td>
</tr>
<tr>
<td>--nobot</td>
<td>Do not scan the boot sector.</td>
<td>page 28</td>
</tr>
<tr>
<td>--nocomp</td>
<td>Same as --ignore-compressed.</td>
<td>page 27</td>
</tr>
<tr>
<td>--nodecrypt</td>
<td>Do not decrypt Microsoft Office compound documents that are password-protected.</td>
<td>page 28</td>
</tr>
<tr>
<td>--nodoc</td>
<td>Do not scan Microsoft Office document files.</td>
<td>page 28</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
<td>See ...</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>--noexpire</td>
<td>Do not issue a warning if the DAT files are out of date.</td>
<td>page 28</td>
</tr>
<tr>
<td>--norename</td>
<td>Do not rename an infected file that cannot be repaired.</td>
<td>page 28</td>
</tr>
<tr>
<td>--noscript</td>
<td>Do not scan files that contain HTML, Javascript, Visual Basic, or Script Component Type Libraries.</td>
<td>page 29</td>
</tr>
<tr>
<td>--one-file-system</td>
<td>Scan an entire directory tree without scanning mounted file systems, if you use this option in conjunction with the --sub option.</td>
<td>page 29</td>
</tr>
<tr>
<td>-p</td>
<td>Same as --atime-preserve.</td>
<td>page 26</td>
</tr>
<tr>
<td>--panalyse</td>
<td>Same as --panalyze.</td>
<td>page 29</td>
</tr>
<tr>
<td>--panalyze</td>
<td>Use heuristic analysis to identify potential program viruses.</td>
<td>page 29</td>
</tr>
<tr>
<td>--plad</td>
<td>Same as --atime-preserve.</td>
<td>page 26</td>
</tr>
<tr>
<td>--program</td>
<td>Scan for malicious applications.</td>
<td>page 29</td>
</tr>
<tr>
<td>-r</td>
<td>Same as --sub.</td>
<td>page 29</td>
</tr>
<tr>
<td>--recursive</td>
<td>Same as --sub.</td>
<td>page 29</td>
</tr>
<tr>
<td>-s</td>
<td>Same as --selected.</td>
<td>page 30</td>
</tr>
<tr>
<td>--selected</td>
<td>Look for viruses in any file that has execute permissions, and all files that are susceptible to virus infection.</td>
<td>page 30</td>
</tr>
<tr>
<td>--secure</td>
<td>Examine all files, unzip archive files and use heuristic analysis.</td>
<td>page 29</td>
</tr>
<tr>
<td>--sub</td>
<td>Examine any subdirectories in addition to the specified target directory.</td>
<td>page 29</td>
</tr>
<tr>
<td>--summary</td>
<td>Produce a summary of the scan.</td>
<td>page 32</td>
</tr>
<tr>
<td>--unzip</td>
<td>Scan inside archive files, such as those saved in ZIP, LHA, PKarc, ARJ, TAR, CHM, and RAR formats.</td>
<td>page 30</td>
</tr>
<tr>
<td>-v</td>
<td>Same as --verbose.</td>
<td>page 32</td>
</tr>
<tr>
<td>--verbose</td>
<td>Display a progress summary during the scan.</td>
<td>page 32</td>
</tr>
<tr>
<td>--version</td>
<td>Display the program’s version number.</td>
<td>page 32</td>
</tr>
<tr>
<td>--virus-list</td>
<td>Display a list of all the viruses that the program can detect.</td>
<td>page 32</td>
</tr>
</tbody>
</table>
Exit codes

When it exits, VirusScan for UNIX returns a code to identify any viruses or problems that were found during a scan.

Table 3-6. Exit codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The scanner found no viruses and returned no errors.</td>
</tr>
<tr>
<td>2</td>
<td>Integrity check on a DAT file failed.</td>
</tr>
<tr>
<td>6</td>
<td>A general problem occurred.</td>
</tr>
<tr>
<td>8</td>
<td>The scanner could not find a DAT file.</td>
</tr>
<tr>
<td>12</td>
<td>The scanner tried to clean a file, and that attempt failed for some reason, and the file is still infected.</td>
</tr>
<tr>
<td>13</td>
<td>The scanner found one or more viruses or hostile objects — such as a Trojan horse, joke program, or test file.</td>
</tr>
<tr>
<td>15</td>
<td>The scanner’s self-check failed; it may be infected or damaged.</td>
</tr>
<tr>
<td>19</td>
<td>The scanner succeeded in cleaning all infected files.</td>
</tr>
<tr>
<td>102</td>
<td>The user quit using the \texttt{--exit-on-error} option.</td>
</tr>
</tbody>
</table>

This code appears when the scan encounters an unexpected condition; for example, if it cannot open a file or runs out of available memory. The program exits immediately and does not finish the scan. This code occurs only if you specified the \texttt{--exit-on-error} option when you started the program. If you did not specify the \texttt{--exit-on-error} option, the scanner returns exit code 6.
VirusScan for UNIX is an effective tool for preventing virus infections, and it is most effective when used in conjunction with regular backups, meaningful password protection, user training, and awareness of virus threats.

To create a secure system environment and minimize the chance of infection, we recommend that you do the following:

- Install VirusScan for UNIX software and other McAfee anti-virus software.
- Include a `uvscan` command in a `crontab` file.
- Make frequent backups of important files. Even if you have VirusScan for UNIX software to prevent attacks from viruses, damage from fire, theft, or vandalism can render your data unrecoverable without a recent backup.

**Detecting new and unidentified viruses**

To offer the best virus protection possible, we continually update the virus definition (DAT) files that the VirusScan for UNIX software uses to detect viruses. For maximum protection, you should regularly update these files.

The term “update” refers only to the DAT files; the term “upgrade” refers to product version revisions, executables, and definition files. We offer free online DAT file updates for the life of your product, but cannot guarantee they will be compatible with previous versions. By upgrading your software to the latest product version and updating regularly to the latest DAT files, you ensure complete virus protection for the term of your software subscription or maintenance plan.
Why do I need new DAT files?

Hundreds of new viruses appear each month. Often, older DAT files cannot assist the VirusScan for UNIX software in detecting these new variations. For example, the DAT files with your original copy of VirusScan for UNIX might not detect a virus that was discovered after you bought the product.

If you suspect you have found a new virus, contact Network Associates.

Updating your DAT files

Download the new files from either of these sources:

- **The Network Associates FTP server.** Open a connection to ftp.nai.com.
  
  Use anonymous as your user name and your e-mail address as your password to gain access. Look for a compressed file in the directory `pub/antivirus/datfiles/4.x`. The file has the format `dat-nnnn.zip`, where `nnnn` is the DAT version number. For example: `dat-4299.zip`.

- **The Network Associates Web Site.** Start your browser, then go to `www.mcafeeb2b.com/naicommon/download/dats/find.asp` to download the latest DAT files.

The number given to the DAT file changes on a regular basis. A higher number indicates a later version of the DAT file.

To use the new DAT files:

1. Create a download directory.
2. Change to the download directory and download the new DAT file from the source you have chosen.
3. To unpack the DAT file, enter the command:
   ```
   tar -xf <file>
   ```
   Here, `<file>` is the name of the file you downloaded.
4. Enter this command to move the DAT files to the directory where your software is installed. Name the file using lower case.
   ```
   mv *.dat /usr/local/uvscan
   ```
   Your system overwrites the old DAT files with the new files. Your software will now use the new DAT files to scan for viruses.
Sample update script for UNIX

The following script is provided only as a suggestion, for you to use and modify to suit your own purposes. It has not been thoroughly tested. Further error checking and password authentication might be required.

The following example shows an update script that gets new DAT files from the Network Associates FTP site.

This entry must appear in the `.netrc` file for this script to work:

```
machine ftp.nai.com
login anonymous
password <e-mail address>
macdef init
cd pub/antivirus/datfiles/4.x
bin
prompt
mget dat-*.tar
close
bye
```

where `<e-mail address>` is the address of the user who is logging in to the FTP server.

```
#!/bin/sh

# Assume uvscan is installed in the same directory as this script.
install_directory=`dirname $0`

# Create a download directory
mkdir /tmp/dat-updates
cd /tmp/dat-updates

# Get the version of the currently installed DAT files
# from the info given by the --version option
current_version=`$install_directory/uvscan --version | grep "Virus data file" | awk '{ print substr($4,2,4) }'`

# Get the new DATs.
# The entry in your .netrc file should take care
# of the downloading.
ftp ftp.nai.com

# Get the version of the new DATs from the filename.
ew_version=`echo dat-*.tar | awk '{ print substr($1,5,4) }'`
```
Preventing Virus Infection

# If they are the same age or older
# than the current ones, do not install them.
if [ "$current_version" -ge "$new_version" ]
  then
  echo "No new DATs available at this time"
  echo "Currently installed version: $current_version"
  echo "Version on FTP site: $new_version"
else
  tar -xf dat-*.tar
  # Move them to the install directory, making sure
  # that the filename is lower case.
  for file in `tar -tf dat-*.tar`
    do
      newfile=`echo $file | tr [A-Z] [a-z]`
      mv ./$file "$install_directory/$newfile"
    done
  # Get the current version again and make sure
  # that the new DATs installed correctly.
  current_version=`
  $install_directory/uvscan --version |
  grep "Virus data file" |
  awk '{ print substr($4,2,4) }'`
  if [ ! "$current_version" -eq "$new_version" ]
    then
      echo "DAT file updates did not work correctly."
      echo "Please try manually."
    fi
  fi
  # Delete the directory that you created.
  cd /
  rm -fr /tmp/dat-updates
Sample update script for Perl

This script is provided only as a suggestion for you to use and modify to suit your own purposes. It has not been thoroughly tested. Further error checking and password authentication might be required.

```perl
#!/usr/bin/perl -w
# uvscan virus DAT file updater written by
# Michael Matsumura (michael+uvscan@limit.org)
# Version 1.0
#
# Net::FTP is required for operation
# and 'tar' should be in the PATH
use strict;
# Set to the directory uvscan is located/installed in.
my $uvscan_directory = "/usr/local/uvscan";

# Set the temporary directory to download
# the DAT archive.
my $tempdir = "/tmp/dat-updates";

# Set to email address for anonymous FTP login
my $emailaddress = "root@";

use Net::FTP;

# Define global variables
my ($ftp, @dirlist, $arraywalk, $localver, $serverver, $localfile, @files, $file);

# Get the local uvscan datfile version
$localver = &checkuvscanver;
print "Currently installed version: \n";

# Create FTP connection
$ftp = Net::FTP->new("ftp.nai.com", Debug => 0);

# Login
$ftp->login("anonymous", $emailaddress);
$ftp->cwd("/pub/antivirus/datfiles/4.x");
$ftp->binary();
@dirlist = $ftp->ls();

foreach $arraywalk (@dirlist) {
    if ($arraywalk =~ /dat-(\[0-9]+)\..tar/i) {
        $serverver = $1;
        print "Version on ftp.nai.com: \n";
        if ($serverver > $localver) {
            print "Updating virus data files...\n";

        } else {
            print "No update available\n";
        }
    }
}
```

Preventing Virus Infection
chdir $tempdir or die("ERROR: Couldn't change directory to tempdir: $tempdir");

# Download the DAT file!
$localfile = $ftp->get($arraywalk);
print "Download complete...updating now\n"

# Untar the files, store the names of them into an array
@files = `tar -xvf $arraywalk`;
foreach $file (@files) {
# A line break is at the end of each $file...
# chomp that off
    chomp($file);
    # Move each file to the uvscan_directory;
    # and make sure they are lowercase.
    my $movestring = "mv $file "$uvscan_directory":".lc($file):
    print "  "$movestring."\n"
    system($movestring);
}

# Make sure that the installation worked,
# by checking if the virus scanner reports
# the same data file version as the one we downloaded.
if (&checkuvscanver eq $serverver) {
    print "Installation successful\n"
} else {
    print "Error in installation, please install manually\n"
}

# Cleanup...
print "Cleaning up\n"

# Remove downloaded DAT archive
unlink($arraywalk) or die("ERROR: Couldn't delete DAT file: $arraywalk")

# Change to filesys root
# and remove temporary directory
chdir("/");
rmdir($tempdir) or die("ERROR: Couldn't remove tempdir: $tempdir");

} else {

if ($serverver > $localver) {
    print "DAT files are the same..no need to update\n";
}
# Don't want to continue if there is more than
# one 'dat-[0-9]+.tar' files
    last;
} }
$ftp->quit;

# uvscan --version reports...
# "Virus data file 4229 created Oct 16 2002"
# &checkuvscanver returns the version
# of the data files.
sub checkuvscanver {
    if (`$uvscan_directory/uvscan --version` =~ /Virus data file
v([0-9]+) created/) {
        return $1;
    }
}
Symbols
“-” option, warning, 22

A
audience for this manual, 5
automatic scan, 21
AVERT Anti-Virus Research Site, contacting, 7

B
backup software, 26
beta program, contacting, 7
bloodhound, see heuristic analysis, 22
boot-sector viruses, 17

C
cache sizes, for archives, 26
cleaning infected files, 31
COM2EXE, 27
compressed files, ignore during scans, 27
configuration file, option for loading saved, 26
configuration options, 19
contacting McAfee
     CONTACT file, 6
     list of resources, 7
conventions, command line, 18
cron, UNIX command, 21
crontab files, for automatic scans, 21
Cryptcom, 27
customer service, contacting, 7

diskette scanning, 27
distributions, versions of software, 11
download web site, 7

e
encrypted files, 28
error codes, 36
error messages, 14
Eudora, 28
examples
    configuring scans, 20
    consecutive options, 18
    cron, 21
    installing on Solaris, 12
    reports, 24
    scanning and cleaning, 21 to 22
    scheduling scans, 21
    --summary option, 24
    update script for Perl, 41
    update script for UNIX, 39
    --verbose option, 24
exit codes, 36
exit-on-error, setting for scans, 27
extra.dat, 27

F
files, list of types scanned, 32

G
general options, 32
getting more information, 6
GZIP, 27

DAT file updates, web site, 7
DAT files, 38
    do not show expiration notice, 28
    updates, 38
diskette scanning, 27
distributions, versions of software, 11
download web site, 7

deleted malware items, 28

examples
    configuring scans, 20
    consecutive options, 18
    cron, 21
    installing on Solaris, 12
    reports, 24
    scanning and cleaning, 21 to 22
    scheduling scans, 21
    --summary option, 24
    update script for Perl, 41
    update script for UNIX, 39
    --verbose option, 24
exit codes, 36
exit-on-error, setting for scans, 27
extra.dat, 27

files, list of types scanned, 32

general options, 32
getting more information, 6
GZIP, 27
Index

H
HELP application, 6
help, online, 18, 32
heuristic analysis, 26, 28 to 29, 31
HTA, 29
HTML, 29

I
IDE, see DAT files, 38
infected files
  cannot be cleaned, 23
  cleaning, 31
renaming, 23
installation requirements, 12
installing VirusScan software, 12
introducing VirusScan, 9

J
Javascript, 29

K
KnowledgeCenter, 7

L
last access date of files, preserving, 26
LIBC6 on Linux, 15
library paths, 13
links, creating to uvscan and shared library, 13
Linux, LIBC5 and LIBC6, 15
list of viruses, 32

M
macros, delete from files, 31
mailboxes, plain text, 28
Matsumura, 41
Microsoft Expand, 27
Microsoft Word files, do not scan, 28
MIME, 28

N
Netscape, 28
new features, 10

O
on-demand scanning, 17
options
  -, 32
  alphabetic list of, 33
  examples, 20 to 22
  general, 32
  report, 24
  response, 21, 31
  scanning, 26
overview of features, 9

P
password crackers, 29
password cracking, 28
pattern files, see DAT files, 38
Perl, 41
permissions, 12
PINE, 28
PKLITE, 27
plain-text mailboxes, 28
preventing virus infection, 37
product training, contacting, 7
progress of scan, 24
progress summary, 32

Q
quarantine, moving infected files to, 31

R
README file, 6
recursion, 29
removing the software
  by hand, 16
  with the uninstallation script, 16
reports, 24
response options, 21, 31
return values, 36
root account, 12
S
scan results, displaying, 32
scan targets, supplying by a file, 27
scan task, 19
scanning
  boot sector of diskette, 27
diskette, 27
  on-demand, 17
  options, 26
  secure, 29
  with maximum security, 19
scheduling a scan, 21
Script Component Type Libraries, 29
secure scanning, 29
shared library path, removing, 16
standard input, to set scan targets, 27
subdirectories, scanning of, 29
summary of scan, 24
summary of scan results, displaying, 32
switches, see options
syntax, variables in, 25
system requirements, 12
T
technical support, contacting, 7
Teledisk, 27
training web site, 7
troubleshooting installation, 14
U
updates, 17
  definition of, 37
upgrade web site, 7
upgrades, definition of, 37
uvscan.exe (VirusScan executable), 21
V
variables, in command line, 25
verbose scan reports, setting, 32
version number, 18, 32
virus definitions, see DAT files, 38
virus signature, 22
viruses
  cleaning infected files, 31
  list of detected, 18
  obtaining a list of, 32
Visual Basic, 29
W
why use VirusScan for UNIX software?, 9
Z
zipped files, ignore during scans, 27