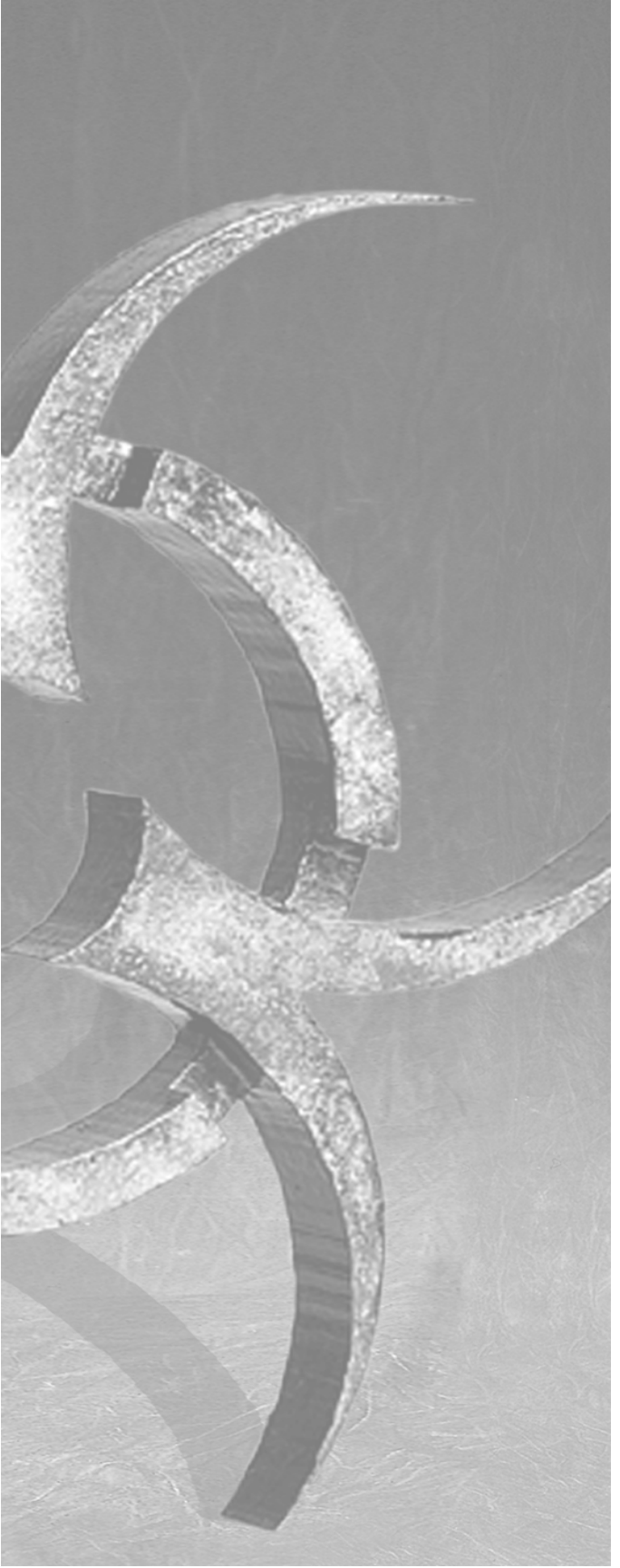

VirusScan for UNIX

Product Guide

Version 4.16.0

McAfee

A Network Associates Company

Table of Contents

Preface

Purpose

This Product Guide provides the following information: descriptions of all product features, instructions for configuring and deploying the software, and procedures for performing tasks. It also provides a roadmap for getting additional information or help.

Audience

This guide is designed for people with some responsibility for configuring and using the software on their own workstations.

Getting more information

HELP	Additional information about the product available in the Help system that is included in the application. To access Help topics, use the man pages in the application.
README.TXT	Product information, system requirements, resolved issues, any known issues, and last-minute additions or changes to the product or this guide.
CONTACT.TXT	A list of phone numbers, street addresses, web addresses, and fax numbers for Network Associates offices in the United States and around the world. It also includes 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 • Resellers
LICENSE.TXT	The terms under which you may use the product. Read it carefully. If you install the product, you agree to the license terms.

Contacting McAfee and Network Associates

Technical Support	http://knowledge.nai.com
Product Documentation	tvd_documentation@nai.com
McAfee Beta Site	www.mcafeeb2b.com/beta/
AVERT Anti-Virus Research Site	www.mcafeeb2b.com/avert
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
Finding a Reseller	www.mcafeeb2b.com/naicommon/partners/tsp-seek/intro.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.TXT file that accompanied this product release.

What is VirusScan for UNIX? 1

VirusScan for UNIX detects and removes viruses on UNIX-based systems.

McAfee anti-virus researchers provide fast, responsive coverage for new viruses and other malicious software. New-generation VirusScan scanning technology can detect and remove more than 58,000 known boot, file, macro, multi-partite, stealth, encrypted, and polymorphic viruses.

A pre-eminent, worldwide staff backs each new update of the virus-scanning engine and release of virus definition .DAT files.

McAfee worldwide virus research team develops weekly updates for the VirusScan virus definition .DAT files, leaving you confident that your network is well protected from attack.

Why use VirusScan for UNIX?

The UNIX operating system is a secure environment, relatively unaffected by computer viruses. The DOS and Windows environment, however, is different. DOS computers have no security and are very susceptible to virus infections. Because DOS system viruses don't affect UNIX systems, you might ask: "Why should I be concerned?"

One reason for concern is that DOS- and Windows-based computers are rapidly appearing on the Internet—and most of these computers use the Internet for file transfer. A UNIX server might still harbor DOS system viruses and, while not itself affected, can pass them on to numerous DOS- and Windows-based clients. Rather than trying to block viruses at each DOS- and Windows-based computer connected to a UNIX system, you can install the VirusScan for UNIX software and use it as an efficient centralized solution. In order to protect yourself and your users, it is more important than ever to maintain anti-virus security.

VirusScan for UNIX software provides the best available anti-virus security, but is only one important element of a comprehensive security program that includes safety measures such as regular backups, meaningful password protection, and training and awareness programs about virus issues.

What comes with the VirusScan for UNIX software?

The VirusScan documentation includes:

- **Product Guide**. This Guide describes how to use the software. It includes background information and advanced configuration options, and is in Adobe Acrobat PDF format. Acrobat PDF files are flexible online documents that contain hyperlinks, outlines, and other aids for easy navigation and information retrieval.
- **README.TXT file**. The README.TXT file contains last-minute additions or changes to the documentation. It lists any known behavior or other issues with the product release, and often describes new product features incorporated into product updates. You can open and print the README.TXT file from Microsoft Windows Notepad, or from nearly any word-processing software.
- **LICENSE.TXT file**. This file outlines the terms of your license to use the software. Read it carefully. By installing the software you agree to its terms.
- **CONTACT.TXT** file. This file contains a list of McAfee addresses and telephone numbers.

Installing VirusScan 2

Before you begin

McAfee distributes the VirusScan software in two ways:

- As an archived file that you can download from the Network Associates web site or from other electronic services.
- On a CD-ROM.

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

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

About the distributions

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

- Sun Microsystems Solaris for SPARC architecture, v2.5.1, 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 v4.2.1, 4.3.x, and AIX 5.0L, with all recommended patches installed.
- Linux v2.x kernels on Intel-based systems - - - Linux v2.x kernels up to and including v2.4 kernels with libc6 (glibc) and the stdc++ library v2.8.

  We do not recommend using VirusScan for UNIX with any Linux development kernel (v2.1.x and v2.3.x). These kernels are built for testing purposes only and are not supplied with any major Linux distribution.
- Santa Cruz Operation (SCO) OpenServer Release 5 and SCO UnixWare 7.1.1.

  Use of VirusScan for UNIX with SCO Openserver requires installation of the SCO UnixWare Binary Compatability Module (BCM).
- FreeBSD v3.2 and 4.3.

If you install VirusScan software from CD-ROM, you will find each version in its own directory. Each distribution has its own installation script.

Installation requirements

To install and run the software, you need:

- The correct version of the UNIX distribution that you require, installed and running correctly on the target machine. See "About the distributions" for information.
- 4MB of free hard disk space for a full installation.
- A CD-ROM 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 anti-virus .DAT files that we offers 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 `vsun4110.tar.Z`) where the example specifies the distribution file.

To start the VirusScan installation script:

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

   If you are using the McAfee installation CD-ROM to obtain the software, you can mount the CD-ROM on to the filesystem.

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

   NOTE: We recommend that you do not copy this file to a separate directory from which you plan to install the program.

3. Type this line at the command prompt to decompress the file to your hard disk:

   `zcat <distribution file> | tar -xf -`

4. Type 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. Do not type the square brackets shown in the command example.

   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. Type **Y** to create each link, or **N** to skip the step.

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

Variable	Distribution
LD_LIBRARY_PATH	Solaris
	SCO OpenServer Release 5
	Linux
	FreeBSD
SHLIB_PATH	HP-UX
LIBPATH	AIX

   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 "When the scanner detects a virus" on page 24 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 2-1. Error messages

Error	Cause or action
Failed to create install_dir	Verify that you have permission to create the installation directory.
Cannot write to install_dir	Verify that you have permission to create installation directory.
The install_dir exists, but is not a subdirectory	Choose another installation directory.
<*file*> is missing	The file might not exist.
<*file*> is not correct	The file did not install correctly.

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}$EICAR-STANDARD-
   ANTIVIRUS-TEST-FILE!$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. Type 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.

> **IMPORTANT:** 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

VirusScan might produce an error message such as: `Cannot find shared object`. The table explains the system-specific reason.

Table 2-2. Error messages

Platform	Cause of error
AIX	The wrong version of the -xlC.rte is installed. VirusScan software will not run on versions before 4.0
Free BSD (Berkeley)	No problems are expected on this platform.
HP-UX	The aCC run-time patch is not installed.
Linux	LIBC6 is supported, but LIBC5 is not.
SCO	No problems are expected on this platform.
Solaris	No problems are expected on this platform.

Table 2-3. Program messages

Program message	Remedy
Unable to find shared library	Set the appropriate environment variable: AIX: LIBPATH HP-UX: SHLIB_PATH SCO OpenServer:LD_LIBRARY_PATH Solaris: LD_LIBRARY_PATH FreeBSD: LD_LIBRARY_PATH Linux: LD_LIBRARY_PATH
Cannot execute: permission denied	Incorrect file permissions can prevent the scanner running correctly. All executables (including the shared libraries) must have Read and Execute permissions (`r_x`), but we recommend `rwxr_xr_x` All .DAT files must have read permissions.
Missing or invalid .DAT files	Re-install the .DAT files.
The program has been altered; please replace with a good copy	The program might be infected. Re-install from the original media.

Removing the program

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

To remove VirusScan software from your system:

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

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

2. When the VirusScan software has been removed, 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 VirusScan program and a shared library path when you installed the VirusScan software, you will need to remove those links yourself.

As the administrator, ensure that your users cannot accidentally remove their VirusScan software.

Removing VirusScan 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.

Using VirusScan 3

VirusScan is a program that 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 operation immediately or schedule automatic scan operations.
- Advanced heuristic scanning detects previously unknown macro 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.

Syntax

This section includes a summary of the command line and its associated options. For a full description of each command, see "Choosing the options" on page 29.

❑ **NOTE:** Some of the options in the syntax summary have a verbose form and an abbreviated form. The abbreviated form appears first, and the verbose form appears after the | symbol. You can use either form to add an option to the command line.

```
uvscan
[--allole] [--analyze,--analyse]
[-c |--clean] [--cleandocall] [--config file]
[--dam] [--dat] [-d |--data-directory directory] [--delete]
[--exclude file] [-e |--exit-on-error]
[--extensions EXT1,[EXT2]] [--extra file]
[--fam] [-f |--file file]
[--floppya], [--floppyb]
[-h |--help] [--ignore-compressed]
[--ignore-links] [--load file]
[--manalyze,--manalyse,--macro-heuristics]
[--maxfilesize X] [--mime] [-m |--move directory]
[--noboot] [--nocomp] [--nodecrypt] [--nodoc] [--noexpire]
[--norename][--noscript][--one-file-system]
[--panalyze,--panalyse][-p,--atime-preserve,--plad]
[-r |--recursive,--sub]
[--secure] [-s|--selected] [--summary]
[--unzip] [-v|--verbose] [--version] [--virus-list]
[-] {file / directory}
```

NOTE: Do not type the square brackets [ ] or the | symbol when you type your options in the command line.

Performing on-demand scan operations

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

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

Choose from three types of options:

- **Scanning options**. These determine how and where the scanner looks for infected files.
- **Response options**. These determine how the scanner responds to any infected files.
- **General options.** These determine how the scanner reports its scanning activities.

Each type of option appears in its own table with a full description of its function. See "Choosing the options" on page 29 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 switches as one switch. For example, rather than typing this:

  ```
  -c -r --one-file-system
  ```

  you can type this instead:

  ```
  -cr --one-file-system
  ```

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

  ```
  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 "Preconfiguring scan operations" on page 22.

❑ **NOTE:** By default, the VirusScan program examines all files, no matter what their extensions. You can limit your scan operation by adding only those extensions you want to examine to the command line after the `--extensions` option, or you may exclude certain files from scan operations with the `--exclude` option. See "Choosing the options" on page 29 for details.

General hints and tips

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

  ```
  uvscan --help
  ```

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

  ```
  uvscan --virus-list
  ```

- To display information about the version of the program, type:

  ```
  uvscan --version
  ```

- To ensure maximum protection from virus attack, you must regularly update your .DAT files. See Appendix A, "Preventing Virus Infection," for details.

Preconfiguring scan operations

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

That way, you can run complete scan operations with ease, and at any time. Your scan task specifies the actions to take when a virus is detected.

To preconfigure a scan operation:

1. Choose the command options you want to use.

   See "Choosing the options" on page 29 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 your text in a file.

4. Type either of these lines at the command prompt.

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

   or

   ```
   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.

5. Press the **RETURN** or **ENTER** key on your keyboard to run the scan operation.

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

> **NOTE:** To learn how to specify the options you want to use, see "Command-line conventions" on page 21.

Example 1

To scan files in the /usr/dos directory according to the settings you stored in the file, /usr/local/config1, type:

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

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

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

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

As an alternative, the contents of the task file can be arranged as single lines:

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

Example 2

To scan only files smaller than 4 Mb and to ignore any compressed files in three separate directories, type:

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

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

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

The contents of the other file, /usr/local/biglist, are:

```
/usr/local/bin
/temp
/etc
```

Scheduling a virus scan operation

You can use the UNIX `cron` scheduler to run automated scan operations. `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 operation 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 operation 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 operation 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 /usr/local/conf1 /uz
```

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

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

When the scanner detects a virus

If the scanner discovers a virus while scanning, it returns exit code number 13. See "Exit codes" on page 27 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 scan operations using one or more of these options:

- `-c or --clean`

  The scanner tries to automatically remove any viruses from infected files. We recommend that you perform another scan operation after using this option to ensure that there are no more viruses in files that the scanner has cleaned.

- `--cleandocall`

  The scanner removes all macros from any file that the program identified as being infected.

- `-m <directory> or --move <directory>`

  The scanner moves any infected files it detects to a quarantine location that you specify. If you use the `-m <directory>` command with `-c`, the scanner copies the infected files to a quarantine location and attempts to clean the original. If the scanner can not clean the original, the file is deleted.

- `--delete`

  The scanner deletes any infected files it finds.

These options are described in detail in the "Response options" table on page 33.

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, type:

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

The VirusScan program (uvscan.exe) scans /usr/dos and its subdirectories automatically, and cleans any infected files it encounters.

Example 2

To scan and clean all files in the /usr/dos directory and its subdirectories, but to ignore any other file systems that are mounted, type:

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

The VirusScan program scans without moving across file systems, and cleans any infected files it detects.

Example 3

To scan all files, except compressed files, in the /usr/dos directory and its subdirectories and to move any infected files to /usr/local/viruses, type:

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

The VirusScan program scans the /usr/dos directory and its subdirectories automatically. It ignores any compressed files. It moves any infected files to /usr/local/viruses.

Example 4

To scan a file with a name prefixed with "-"., type:

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

The VirusScan 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
```

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 3-1. Renaming infected files

Original	Renamed	Description
Not v??	v??	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*.
v??	vir	File extensions that start with *v* are renamed as *.vir*. For example, *myfile.vbs* becomes *myfile.vir*.
vir, v01-v99		These files are recognised as already infected, and are not renamed again.
<blank>	vir	Files with no extensions are given the extension, *.vir*.

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.

Exit codes

The VirusScan program returns an code when it exits. These codes identify any viruses or problems that were found during a scan operation.

Table 3-2. Exit codes

Code number	Description
0	The scanner found no viruses and returned no errors.
2	Driver integrity check failed.
6	A general problem occurred.
8	The scanner could not find a driver.
12	The scanner tried to clean a file, and that attempt failed for some reason, and the file is still infected.
13	The scanner found one or more viruses or hostile objects (such as a Trojan horse, joke, or a test file).
15	The scanner's self-check failed; it may be infected or damaged.
19	The scanner succeeded in cleaning all infected files.
102	The user quit using the `--exit-on-error` option. This code appears when the scan operation 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 operation. This code occurs only if you specified the `--exit-on-error` option when you started the program. If you did not specify the `--exit-on-error` option, the scanner returns exit code 6.

Report features

The program can take some time to complete a scan operation, 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 33.

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

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

- How many files the program scanned,
- How many files it cleaned,
- How many files it did not scan, and
- How many infected files it found.

Example

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

```
$ uvscan --summary  -v /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 tables describe the options you can use to target your scan operation. The descriptions use these conventions to identify the options or required variables:

- 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 21.

Scanning options

Scanning options describe how and where each scan operation will look for infected files. You can use a combination of these options to customize the scan operation to suit your needs. Most options are off by default. If the option is normally on, the option runs automatically; you do not need to add it to the command line. You can override some of these default options with other options.

Table 3-3. Scanning options

Option	Description
`--allole`	Check every file for OLE objects.
`--analyze`, `--analyse`	Use heuristics to find possible viruses in "clean" files. This step occurs after the program has checked for other viruses.
`--config <file>`, `--load <file>`	Run the options specified in `<file>`. You may not nest configuration files within other configuration files.
`-d <directory>`, `--dat <directory>` `--data-directory <directory>`	Specify where to find SCAN.DAT, NAMES.DAT, and CLEAN.DAT. If the `-d` switch is not used 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.
`--exclude <file>`	Exclude the directories or files from the scan operation as specified in `<file>`.

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

Option	Description
`-e, --exit-on-error`	Quit and display an error message if an error is found. The error message indicates the severity of the error. See page 27 for an explanation of exit codes.
`--extensions <EXT1[,EXT2,...]>`	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, it launches the "`-s, --selected`" option.
`--extra <file>`	Specify where to find EXTRA.DAT. If this switch is not used 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.
`-f <file>, --file <file>`	Scan the directories or files as specified in `<file>`.
`--floppya, --floppyb`	Scan the boot sector of the 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 also reduces file security. By default, the program scans compressed files.
`--ignore-links`	Do not resolve any symbolic links and do not scan the link targets.
`--load <file>`	See `--config` option.

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

Option	Description
`--manalyze, --manalyse` `--macro-heuristics`	Use heuristic detection to identify potential macro viruses. This is a subset of "`--analyze, --analyse`".
`--maxfilesize X`	Examine only those files smaller than *X* size. Here, *X* is a file size measured in megabytes.
`--mime`	Scan MIME-encoded files.
`--noboot`	Do not scan the boot-sector on startup.
`--nodecrypt`	Do not decrypt encrypted files in order to scan them.
`--noexpire`	Do not issue a warning if the .DAT files are out of date.
`--nodoc`	Do not scan Microsoft Office document files.
`--norename`	Do not rename an infected file that cannot be repaired. See "Handling an infected file that cannot be cleaned" on page 26 for details about renaming.
`--noscript`	Do not scan files that contain HTML, Javascript, Visual Basic, or Script Component Type Libraries. Stand-alone Javascript and Visual Basic Script files will still be scanned.
`--one-file-system`	Scan an entire directory tree without scanning mounted file systems, if you use this option in conjunction with the "`-r, --recursive, --sub`" option. Normally, the program treats a mount point as a subdirectory and scans that file system. This option prevents the scan operation from running in subdirectories that are on a different file system to the original directory.
`--panalyze, --panalyse`	Use heuristic detection to identify potential program viruses. This is a subset of "`--analyze, --analyse`".

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

Option	Description
`-p, --atime-preserve, --plad`	Reset the time that the file was last accessed to what it was before the scan operation. This enables backup software (which relies on this date) to run correctly. The scanner will not reset the access time if the file contained a virus or if the person who starts the scan operation does not own the file.
`--program`	Scan for malicious applications. Some widely available applications (such as “password crackers”) can be used maliciously or can pose a security threat.
`-r, --recursive, --sub`	Examine any specified target directory and all its subdirectories.
`--secure`	Examine all files. This option activates the `--analyze` and `--unzip` options and deactivates the `--selected` and `--extensions` options at the same time.
`-s, --selected`	Look for viruses in any file that has execute permissions, and all files that are susceptible to virus infection. By default, this option is off. Over 80 file types are scanned including .EXE, .COM, .BAT, .DOT, .DOC, .BIN, and .VBS. By scanning only files which are susceptible to virus infection, the program can scan a directory faster.
`--unzip`	Scan inside archive files, including those saved in ZIP, LHA, PKarc, ARJ, TAR, CHM, CAB and RAR formats. If used with --clean, this option attempts to clean non-compressed files inside ZIP files only. No other archive formats are supported for cleaning. --clean will not delete infected files within ZIPs nor will it rename infected files within ZIPs. --clean will also not rename the ZIP file itself. If any files are identified within any other archive format that cannot be cleaned you must first extract them from the archive file.

Response options

These options determine how your scan operation responds to a virus infection. You can use a combination of these options to customize the scan operation. None of the options in this table occurs automatically. To activate each option, specify it in your command line. You may override some of these default options with other options.

Table 3-4. Response options

Option	Description
`-c, --clean`	Try automatically to remove any viruses from infected files. If the program cannot clean the file, it displays a warning message. If you use this option, repeat the scan operation to ensure there are no more infections.
`--cleandocall` `--dam`	Delete all macros from a potentially infected file. Use these options with caution when you also use the `"--fam"` option.
`--delete`	Automatically delete any infected files that are found.
`-m <directory>,` `--move <directory>`	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 the `-m <directory>` command with `-c`, the program copies the infected files to a quarantine location and attempts to clean the original. If it can't clean the original, the file is deleted.

General options

These options provide help or give you additional information about the scan operation. You may use a combination of these options to customize the scan operation. None of the options in this table occur automatically. To activate each option, specify it as part of your command line. You may override some of these default options with other options.

Table 3-5. General options

Option	Description
`-`	Denote the end of the options and the start of the target to be scanned. This is optional. This feature is particularly useful with file names that are prefixed with "-", because it avoids confusion between the options and the target.
`--extlist`	Display a list of all file extensions which are susceptible to virus infection; that is, those that are scanned when `-s` or `--selected` is set.
`-h, --help`	List the most commonly used options, with a short description. For a full description, use `man uvscan`.
`--summary`	Produce a summary of the scan operation that includes the following: • How many files the scanner examined. • How many infected files the scanner found. • How many viruses the scanner removed from infected files.
`-v, --verbose`	Display a progress summary during the scan operation.
`--version`	Display the program's version number.
`--virus-list`	Display a list of all the viruses that the program can detect.

Options in alphabetic order

For convenience, the options are repeated in this section in alphabetic order.

Table 3-6. Options in alphabetic order

Option	Description
-	Denote the end of the options and the start of the target to be scanned. This is optional.
--allole	Check every file for OLE objects.
--analyze, --analyse	Use heuristics to find possible viruses in “clean” files.
--atime-preserve	Reset the time that the file was last accessed to what it was before the scan operation.
-c, --clean	Try automatically to remove any viruses from infected files.
--cleandocall	Delete all macros from a potentially infected file.
--config <file>	Run the options specified in <file>.
-d <directory>, --dat <directory> --data-directory <directory>	Specify where to find SCAN.DAT, NAMES.DAT, and CLEAN.DAT.
--dam	See --cleandocall.
--delete	Automatically delete any infected files that are found.
-e, --exit-on-error	Quit and display an error message if an error is found.
--exclude <file>	Exclude the directories or files from the scan operation as specified in <file>.
--extensions <EXT1[,EXT2,...]>	Examine files that have the specified extension.
--extlist	Display a list of all file extensions which are susceptible to virus infection; that is, those that are scanned when -s or --selected is set.
--extra <file>	Specify where to find EXTRA.DAT.
-f <file>, --file <file>	Scan the directories or files as specified in <file>.

Table 3-6. Options in alphabetic order (*Continued*)

Option	Description
`--fam`	Locate all files that have macros.
`--floppya, --floppyb`	Scan the boot sector of the floppy disk in drive A or B.
`-h, --help`	List the most commonly used options, with a short description. For a full description, use man uvscan.
`--ignore-compressed`	Ignore compressed files.
`--ignore-links`	Do not resolve any symbolic links and do not scan the link targets.
`--load <file>`	See --config option.
`-m <directory>`	Move any infected files to a quarantine location as specified.
`--macro-heuristics, --manalyze, --manalyse`	Use heuristic detection to identify potential macro viruses.
`--maxfilesize X`	Examine only those files smaller than X size.
`--mime`	Scan MIME-encoded files.
`--move <directory>`	See `-m <directory>`.
`--noboot`	Do not scan the boot-sector on startup.
`--nocomp`	See `--ignore-compressed`.
`--nodecrypt`	Do not decrypt encrypted files in order to scan them.
`--nodoc`	Do not scan Microsoft Office document files.
`--noexpire`	Do not issue a warning if the .DAT files are out of date.
`--norename`	Do not rename an infected file that cannot be repaired.
`--noscript`	Do not scan files that contain HTML, Javascript, Visual Basic, or Script Component Type Libraries.
`--one-file-system`	Scan an entire directory tree without scanning mounted file systems, if you use this option in conjunction with the “-r, --recursive, --sub” option.

Table 3-6. Options in alphabetic order (*Continued*)

Option	Description
`-p, --plad`	See `--atime-preserve`.
`--panalyze, --panalyse`	Use heuristic detection to identify potential program viruses.
`--program`	Scan for malicious applications.
`-r, --recursive`	Examine any specified target directory and all its subdirectories.
`-s, --selected`	Look for viruses in any file that has execute permissions, and all files that are susceptible to virus infection. By default, this option is off.
`--secure`	Examine all files.
`--sub`	See `-r, --recursive`.
`--summary`	Produce a summary of the scan operation that includes the following:
`--unzip`	Scan inside archive files, including those saved in ZIP, LHA, PKarc, ARJ, TAR, CHM, CAB and RAR formats.
`-v, --verbose`	Display a progress summary during the scan operation.
`--version`	Display the program's version number.
`--virus-list`	Display a list of all the viruses that the program can detect.

Preventing Virus Infection A

Creating a secure system environment

VirusScan anti-virus software is an effective tool for preventing virus infections, but 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 your chance of infection, we recommend that you:

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

Detecting new and unidentified viruses

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

❑ **NOTE:** The term “update” refers only to the .DAT files; the term “upgrade” refers to product version revisions, executables, and definition files. McAfee offers 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 a new .DAT file?

Hundreds of new viruses appear each month. Often, older .DAT files cannot assist the VirusScan software in detecting these new variations. For example, the .DAT files with your original copy of VirusScan 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 VirusScan .DAT files in the directory pub/antivirus/datfiles/4.x.

- **The Network Associates Web Site.** Start your browser, then go to http://www.nai.com/download to download the latest .DAT files.

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.

   The number given to the .DAT file will change on a regular basis. A higher number indicates a later version of the .DAT file.
3. To unpack the .DAT file, type:

   ```
   tar -xf <file>
   ```

   Here, `file` is the name of the file you downloaded.
4. Type this line at the command prompt to move the .DAT files to the directory where your software is installed:

   ```
   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.

IMPORTANT: Name the file using lower case.

Sample update script for UNIX

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.

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 .DATs
# from the info given by the --version switch
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.
new_version=`echo dat-*.tar | awk '{ print substr($1,5,4) }'`

# 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
    # 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
    # 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.

```
#!/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 to 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
#
$localver = &checkuvscanver;
print "Currently installed version: ".$localver."\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 = $1;
    print "Version on ftp.nai.com: ".$serverver."\n";
    if ($serverver > $localver) {
      print "Updating virus data files...\n";

      # Create and then change the working dir to $tempdir
      #
      if (!(-d $tempdir)) {
        mkdir($tempdir, 700) or die("ERROR: Couldn't make temporary directory:
$tempdir");
      }
      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 v4119 created Feb 03 2001"
# &checkuvscanver returns the version of the data files
#
sub checkuvscanver {
  if (`$uvscan_directory/uvscan --version` =~ /Virus data file

v([0-9]+) created/) {
    return $1;
  }
}
```

Index

L

M

N

O

P

Q

R

S

T

U

V

W

Z