Linux Security 64

Contents

Chapter 1: Introduction 4

1.1 System requirements..........5

1.2 How the product works 5

1.3 Key features and benefits 6

1.4 What is harmful content..........7

1.4.1 Viruses..........7

1.4.2 Potentially unwanted applications (PUA) and unwanted applications (UA)..........7

1.4.3 Worms..........7

1.4.4 Trojans..........8

1.4.5 Backdoors..........8

1.4.6 Exploits..........8

1.4.7 Exploit kits..........9

1.4.8 Rootkits..........9

Chapter 2: Installation..........10

2.1 Creating the installation package..........11

2.2 Creating the content package for isolated environments..........11

2.3 Deploying the installation package..........13

2.3.1 Delaying product activation..........15

2.4 Uninstalling the product..........15

Chapter 3: Using the product 17

3.1 Real-time scanning..........18

3.2 Integrity checking 18

3.3 Automatic updates 20

3.4 Using HTTP proxies..........20

3.5 Example: managing integrity checker profiles..........21

3.6 Basics of using F-Secure Policy Manager 22

3.6.1 Using real-time scanning with Policy Manager..........22

3.6.2 Using integrity checking with Policy Manager..........25

3.6.3 Configuring automatic update options with Policy Manager..........25

3.7 Configuring settings with the lsctl utility..........27

3.7.1 Setting types and the settings tree..........27

3.7.2 Operating with settings..........29

3.7.3 Input and output formats..........30

3.7.4 Working with arrays..........31

3.7.5 Loading product settings from a file..........32

3.7.6 Using lsctl and Policy Manager..........32

3.7.7 Inspecting the state of the product..........33

3.7.8 Command-line settings for real-time scanning....35
3.7.9 Command-line settings for integrity checking....38
3.7.10 Command-line settings for offload scanning....39
3.7.11 Command-line settings for automatic updates....39
3.8 Command line usage....41
3.8.1 Starting and stopping services....41
3.8.2 Scanning the computer manually from the command line....41
3.8.3 Testing the antivirus protection 43
3.8.4 Running the integrity checker from the command line....43
3.8.5 Updating the product manually from the command line....43
3.8.6 Updating the product manually using an update archive....44

Appendix A: Alert severity levels 46

Appendix B: Services and logs....48

B.1 Services installed with the product....49
B.2 Logs used by the product....50

Appendix C: Cloud services....51

Chapter
1

Introduction

Topics:

- System requirements
- How the product works
- Key features and benefits
- What is harmful content

This product provides an integrated, out-of-the-box security solution with strong real-time protection against viruses and potentially unwanted applications, and it also includes host intrusion prevention (HIPS) functionality that provides protection against unauthorized system modifications, as well as userspace and kernel rootkits.

Computer viruses are one of the most harmful threats to the security of data on computers. While some viruses are harmless pranks, other viruses can destroy data and pose a real threat.

The solution can be easily deployed and managed using WithSecure Policy Manager. It is also possible to run Linux Security 64 without a connection to Policy Manager, in which case you can use command line tools to manage the product settings.

Note: Compared to previous versions, Linux Security 64 no longer includes a separate web user interface.

1.1 System requirements

This section contains important information about the product.

We strongly recommend that you read the entire document before you start using the product.

Supported platforms

Note: WithSecure supports only those operating systems that are supported by their vendor. If you are interested in long-term support for a platform that vendors no longer support, contact your sales representative.

The following 64-bit (AMD64/EM64T) distributions are supported:

- AlmaLinux 8
- AlmaLinux 9
- Amazon Linux 2
- CentOS 7 (7.3 or newer)
- CentOS Stream 8
- Debian 10
- Debian 11
- Oracle Linux 7 (7.3 or newer)
- Oracle Linux 8
- Oracle Linux 9
- RHEL 7 (7.3 or newer)
- RHEL 8
- RHEL 9
- Rocky Linux 8
- Rocky Linux 9
- SUSE Linux Enterprise Server 12 (Service Pack 1 or newer)
- SUSE Linux Enterprise Server 15 (Service Pack 1 or newer)
- Ubuntu 18.04
- Ubuntu 20.04
- Ubuntu 22.04

Support for SELinux

The product supports Security-Enhanced Linux with the following distributions:

- AlmaLinux 8, 9 *
- CentOS 7, 8 *
- CentOS Stream 8 *
- Debian 10, 11 **
- Oracle Linux 7, 8, 9 *
- RHEL 7, 8, 9 *
- Rocky Linux 8, 9 *

* on systems running the "targeted" SELinux system policy

** the product is compatible with the "default" SELinux system policy

If the distribution is not supported, SELinux must be disabled.

1.2 How the product works

The product detects and prevents intrusions and protects against malware.

When user downloads a file from the Internet, for example by clicking a link in an e-mail message, the file is scanned when the user tries to open it. If the file is infected, the product protects the system against the malware.

- ***Real-time scanning*** gives you continuous protection against viruses and potentially unwanted applications as files are opened, copied, and downloaded from the Web. Real-time scanning functions transparently in the background, looking for viruses whenever you access files on the hard disk, removable media, or network drives. If you try to access an infected file, the real-time protection automatically stops the virus from executing.
- When the real-time scanning has been configured to scan a limited set of files, the ***manual scanning*** can be used to scan the full system or you can use the scheduled scanning to scan the full system at regular intervals.
- ***Automatic Updates*** keep the virus definitions always up-to-date. The virus definition databases are updated automatically after the product has been installed. The virus definitions updates are signed by F-Secure.

The Host Intrusion Prevention System (***HIPS***) detects any malicious activity on the host, protecting the system on many levels.

- ***Integrity Checking*** protects the system against unauthorized modifications. It is based on the concept of a known good configuration - the product should be installed before the computer is connected to the network to guarantee that the system is in a known good configuration.

  You can create a baseline of the system files that you want to protect and prevent the use of any modified files for all users.
- If an attacker gains a shell access to the system and tries to add a user account to login to the system later, Host Intrusion Prevention System (***HIPS***) detects modified system files and alerts the administrator.
- If an attacker has gained an access to the system and tries to install a userspace rootkit by replacing various system utilities, ***HIPS*** detects modified system files and alerts the administrator.

1.3 Key features and benefits

The product offers superior protection against viruses and worms and is transparent to end-users.

The product scans files on a wide range of Linux-supported file systems.

- Scans files on any Linux-supported file system.
- Superior detection rate with multiple scanning engines.
- A heuristic scanning engine can detect suspicious, potentially harmful files.
- The product can detect and categorize potentially unwanted applications.
- The product can be configured so that the users cannot bypass the protection.
- Files are scanned for viruses when they are opened or closed and before they are executed.
- You can specify what files to scan, how to scan them, what action to take when malicious content is found and how to alert about the infections.
- Recursive scanning of archive files.
- Virus definition database updates are signed for security.

The product works totally transparently to the end users.

- Virus definition databases are updated automatically without any need for end-user intervention.

Critical information of system files is stored and automatically checked before access is allowed.

- The administrator can protect files against changes so that it is not possible to install, for example, a trojan version of a software.
- An alert is sent to the administrator when a modified system file is found.

The default settings apply in most systems.

- Security policies are configured and distributed from one central location.

The product has extensive monitoring and alerting functions that can be used to notify any administrator in the company network about any infected content that has been found.

1.4 What is harmful content

Harmful applications and files can try to damage your data or gain unauthorized access to your computer system to steal your private information.

1.4.1 Viruses

A virus is usually a program that can attach itself to files and replicate itself repeatedly; it can alter and replace the contents of other files in a way that may damage the computer.

A ***virus*** is a program that is normally installed on the computer without the user's knowledge. Once there, the virus tries to replicate itself. The virus:

- uses some of the system resources
- may alter or damage files on the computer
- tries to use the computer to infect other computers
- may allow the computer to be used for illegal purposes.

1.4.2 Potentially unwanted applications (PUA) and unwanted applications (UA)

'Potentially unwanted applications' have features that you may like or want. 'Unwanted applications' have features that can harm your device or data more seriously.

An application may be called an 'potentially unwanted' (PUA) if it:

- **Hurts your privacy or productivity** - for example, exposes personal information or does things without your permission
- **Uses too many of your device's resources** - for example, takes up a lot of storage space or memory
- **Makes your device or data less secure** - for example, exposes you to unexpected content or other applications

The harm that these features can cause to your device or data can be from mild to serious. However, they are not dangerous enough to be called malware.

If an application has features that cause serious harm, it is labeled as 'unwanted application' (UA). The product will treat such applications with more caution.

The software will treat an application differently depending on whether it is a 'potentially unwanted' (PUA) or an 'unwanted application' (UA):

- **A potentially unwanted application** - The product stops the application from running. If you are sure that you trust the application, you can exclude it from scanning. To do this, you need to have administrative rights on the file to be excluded.
- **An unwanted application** - The product stops the application from running.

1.4.3 Worms

Worms are programs that send copies of themselves from one device to another over a network. Some worms also perform harmful actions on an affected device.

Many worms are designed to appear attractive to a user. They may look like images, videos, applications or any other kind of useful program or file. The aim of the deception is to lure the user into installing the worm. Other worms are designed to be completely stealthy, as they exploit flaws in the device (or in programs installed on it) to install themselves without ever being noticed by the user.

Once installed, the worm uses the device's physical resources to create copies of itself, and then send those copies to any other devices it can reach over a network. If a large quantity of worm copies is being sent out, the device's performance may suffer. If many devices on a network are affected and sending out worm copies, the network itself may be disrupted. Some worms can also do more direct damage to an affected device, such as modifying files stored on it, installing other harmful applications or stealing data.

Most worms only spread over one particular type of network. Some worms can spread over two or more types, though they are relatively rare. Usually, worms will try and spread over one of the following networks (though there are those that target less popular channels):

- Local networks
- Email networks
- Social media sites
- Peer-to-peer (P2P) connections
- SMS or MMS messages

1.4.4 Trojans

A trojan is a program that offers, or appears to offer, an attractive function or feature, but then quietly performs harmful actions in the background.

Named after the Trojan Horse of Greek legend, trojans are designed to appear attractive to a user. They may look like games, screensavers, application updates or any other kind of useful program or file. Some trojans will mimic or even outrightly copy popular or well-known programs to appear more trustworthy. The aim of the deception is to lure the user into installing the trojan.

Once installed, trojans can also use 'decoys' to maintain the illusion that they are legitimate. For example, a trojan disguised as a screensaver application or a document file will display an image or a document. While the user is distracted by these decoys, the trojan can quietly perform other actions in the background.

Trojans will usually either make harmful changes to the device (such as deleting or encrypting files, or changing program settings) or steal confidential data stored on it. Trojans can be grouped by the actions they perform:

- **Trojan-downloader**: connects to a remote site to download and install other programs
- **Trojan-dropper**: contains one or more additional programs, which it installs
- **Trojan-pws**: Steals passwords stored on the device or entered into a web browser
    - **Banking-trojan**: A specialized trojan-pws that specifically looks for usernames and passwords for online banking portals
- **Trojan-spy**: Monitors activity on the device and forwards the details to a remote site

1.4.5 Backdoors

Backdoors are features or specially made software that let someone bypass the security of a specific program, device, website, or service. Usually, they are used by attackers to get into a system or do something harmful without permission.

A feature in a program, device, website, or service can be considered a backdoor if its design or implementation introduces a security risk. For example, a hidden way for an administrator to access a website that always works with the same password could be considered a backdoor.

Backdoor software takes advantage of weaknesses in the code of a targeted program, device, website, or service. These weaknesses could be mistakes in the code, vulnerabilities, or hidden features that are not meant to be used.

Backdoors are usually used by attackers to get into a system without permission or to do something harmful while getting around security features such as limits on who can access files or devices, and to bypass authentication and encryption.

1.4.6 Exploits

Exploits are objects or ways of taking advantage of a flaw in a software to make it behave in a way that it was not meant. This makes it possible for an attacker to cause harm to your computer or device.

An exploit can be either an object or a method. For example, a specially made software, a piece of code, or a string of characters are all objects whereas a specific order of running commands is a method.

An exploit can be used to take advantage of a flaw or weak spot (also known as a vulnerability) in a software. Because each software is different, each exploit has to be made specifically for that software.

There are several ways an attacker can use an exploit to harm your computer or device:

- **Embedding it inside a hacked or specially made software** - when you install and launch the software, the exploit is launched
- **Embedding it inside a document that is sent as an email attachment** - when you open the attached document, the exploit is launched
- **Hosting it on a hacked or harmful website** - when you visit the website, the exploit is launched

Launching the exploit makes the software behave in a strange way, such as making it crash, or changing the system's storage or memory. This makes it possible for someone to harm your device and data, such as stealing your information or getting access to restricted sections of the operating system.

1.4.7 Exploit kits

Exploit kits are toolkits used by attackers to manage exploits and deliver harmful programs to a vulnerable computer or device.

An exploit kit contains an inventory of exploits, each of which can take advantage of a flaw (vulnerability) in a program, computer or device. The kit itself is usually hosted on a harmful or a hacked site, so that any computer or device that visits the site is exposed to its effects.

When a new computer or device connects to the booby-trapped site, the exploit kit probes it for any flaws that can be affected by an exploit in the kit's inventory. If one is found, the kit launches the exploit to take advantage of that vulnerability.

After the computer or device is compromised, the exploit kit can deliver a payload to it. This is usually another harmful program that is installed and launched on the computer or device, which in turn performs other unauthorized actions.

Exploit kits are designed to be modular and easy to use, so that their controllers can simply add or remove exploits and payloads to the toolkit.

1.4.8 Rootkits

Rootkits are programs that make other ***malware*** difficult to find.

Rootkit programs subvert the control of the operating system from its legitimate functions. Usually, a rootkit tries to obscure its installation and prevent its removal by concealing running processes, files or system data from the operating system. In general, rootkits do this to hide malicious activity on the computer.

Protection against userspace rootkits

If an attacker has gained an access to the system and tries to install a userspace rootkit by replacing various system utilities, ***HIPS*** detects modified system files and alerts the administrator.

Chapter 2

Installation

Topics:

- Creating the installation package
- Creating the content package for isolated environments
- Deploying the installation package
- Uninstalling the product

This section contains instructions for installing the product.

Installing Linux Security 64 requires first creating an installation package in F-Secure Policy Manager, and then deploying it on each target computer.

2.1 Creating the installation package

Follow the instructions given here to create an installation package for deployment on the target computers.

Note: To install F-Secure Linux Security 64 - even for use as a stand-alone product - you need to use F-Secure Policy Manager to configure and create the installation package. For more information on using Policy Manager, see the Policy Manager administrator's guide.

Note: Starting from Policy Manager version 15.21, you can bundle the policy with the installer so that initial values for settings, including the pinned version number are taken from Policy Manager. Configure the policy on the host or root level in Policy Manager before you start the process and use the `fsls64-4.0.*.jar` installer.

Note: If you use a pinned version of the product, each pinnable product version is supported for one year from the release date of the subsequent pinnable version in general. Exceptions may be outlined in the change log of the pinnable version release. The pinned version of the product still requires an online internet connection.

1. In Policy Manager Console, select **Tools** > **Installation packages** from the menu.
   This opens the **Installation packages** window.
2. Click **Import**.
3. Select the Linux Security 64 installation package that you want to use, then click **Import**.
4. Select the imported installation package in the packages list and click **Export**.
5. Enter a name and select a folder for the exported `zip` file.
   A **Remote Installation Wizard** window opens.
6. Click **Next**.
7. Choose either **No initial policy** if you do not want to bundle the policy with the installer or select the policy domain or host for the initial policy that is bundled with the installer.
8. Enter your license keycode for the product, then click **Next**.
9. Enter the Policy Manager Server address:
   - If the installation package is for computers that are centrally managed through Policy Manager, enter the address for your Policy Manager Server and modify the ports to use for both HTTP and HTTPS communication if necessary.
   - If the installation package is for stand-alone deployment on computers that are not connected to Policy Manager, enter `0.0.0.0` as the Policy Manager Server address.
10. Click **Finish**.

If you bundle the policy with the installer and export the installer in stand-alone mode, the policy is applied as the initial policy. If you export the installer in the managed mode, Policy Manager will override the initial policy when the host connects to it. To prevent this, use the following command when installing the product:

```
bash ./f-secure-linuxsecurity/f-secure-linuxsecurity-installer
--product-version=[pinned version]
```

This prevents overriding the pinned version setting, but does not prevent overriding the full policy.

2.2 Creating the content package for isolated environments

To install F-Secure Linux Security 64 in an isolated environment with limited network connectivity, you need to prepare an additional content package for the product installer.

Note: To generate the content package, use a computer that has network access to F-Secure servers and either Policy Manager Server (Linux or Windows) installed or a separately available `fspm-definitions-update-tool` downloaded (Linux).

The installation package that you create using Policy Manager Console installs the latest available version of the product by downloading it over the network. Skip these steps if you are only deploying the installation package to hosts that can connect to the network to download data during installation.

The additional package for isolated hosts provides the necessary components that are usually downloaded from the network during installation. Use this with the installation package on each isolated host to which you deploy the product.

Follow these instructions to create the content package.

Note: These instructions are for Linux. You can find the command path and the syntax for Windows in the Policy Manager administrator's guide.

1. Open a command line.
2. Select the directory where you want to extract the Policy Manager definitions update tool.

   Make sure that you use an existing directory. The examples in these steps use `<DIR>` as a placeholder for this directory.
3. Extract the definitions update tool.
   - If you have Policy Manager Server, run the following command:

     ```
     /opt/f-secure/fspms/bin/prepare-fspm-definitions-update-tool <DIR>
     ```
   - If you have downloaded the `fspm-definitions-update-tool.tar.gz` file, run the following command:

     ```
     tar -C <DIR> -xf /PATH/TO/fspm-definitions-update-tool.tar.gz
     ```
4. Download the `channels.json` file to your system from the following location.
5. Copy the file into the `<DIR>/fspm-definitions-update-tool/conf/` directory.

   Note: Make sure to keep the `channels.json` file name when copying. You may safely replace any previous version of the file which may already exist in the directory.
6. Make sure that the `<DIR>/fspm-definitions-update-tool/data/` directory is empty or does not exist by running the following command:

   ```
   rm -rf <DIR>/fspm-definitions-update-tool/data/
   ```
7. Run the following command to create the content package:

   ```
   <DIR>/fspm-definitions-update-tool/fspm-definitions-update-tool
   ```

   The output for this command is similar to the following example:

   ```
   Checking for updates...
   "fmlibunix64" is successfully updated to version "1579786361"
   "fsbg-100-linux-x86_64" is successfully updated to version "1576151869"
   "linuxsecurity-1200-linux-x86_64" is successfully updated to version
   "1576153361"
   "fsbspamd-100-linux-x86_64" is successfully updated to version "1575373406"

   "hydra-linux64" is successfully updated to version "1584976097"
   "baseguard-100-linux-x86_64" is successfully updated to version "1582721318"

   "aqualnx64" is successfully updated to version "1585032882"
   Update check completed successfully, new updates downloaded.
   Malware definitions archive is ready:
   /tmp/fspm-definitions-update-tool/data/f-secure-updates.zip
   ```

   The last line of the output shows the path for the product content package.

Note: To create another content package with newer updates, repeat steps 6 and 7. An empty data directory is required to create packages that are valid for use with F-Secure Linux Security.

2.3 Deploying the installation package

Once you have created the installation package, follow these instructions to deploy it on the target computers.

1. Copy the exported `zip` installation package to the Linux hosts in your network.

   For any isolated hosts, also copy the `zip` content package that includes the components that are normally downloaded during installation.

2. Install the product on each host:

   a) Log in to the Linux host as `root`.

   b) Check that the required dependencies are installed:

   - Amazon 2, CentOS 7, Oracle Linux 7, RHEL 7: `libcurl`, `python`
   - AlmaLinux 8, CentOS Stream 8, Oracle Linux 8, RHEL 8, Rocky Linux 8: `libcurl`, `python36` or `python39`
   - AlmaLinux 9, Oracle Linux 9, RHEL 9 , Rocky Linux 9: `libcurl`, `python3`
   - Debian 10, Ubuntu 18.04: `libcurl4`, `python`
   - Debian 11, Ubuntu 20.04: `libcurl4`, `python3`
   - SUSE Linux Enterprise Server: `libcurl4`, `python3`
   - Ubuntu 22.04: `libcurl4`, `python3`

   c) Extract the installation package that you exported from Policy Manager Console to a fresh, empty directory.

   d) Run the installation command.

   The command to use depends on whether or not the host can connect to F-Secure backend systems over the network.

   - To install the latest version of the product over the network, run:

     ```
     bash f-secure-linuxsecurity/f-secure-linuxsecurity-installer
     ```

     This command installs the latest version of the product and configures it to be updated automatically.

   - To install and keep using a specific version of the product without updating it automatically, use the `--product-version` option to indicate the version that you want in the installer command.

     Note: The `fsls64-3.0.x.jar` installer can only be used with specific product versions starting from "linuxsecurity-2021_2".

   - To install the product on an isolated host using a content package (`f-secure-updates.zip`), run:

     ```
     bash f-secure-linuxsecurity/f-secure-linuxsecurity-installer
     --package=/PATH/TO/f-secure-updates.zip --automatic-updates=none
     ```

     On an isolated host, the product does not download or install any product or malware definition database updates automatically. If you want to install the product using a content package but still let the product fetch updates automatically over the network, remove the `--automatic-updates=none` option from the `f-secure-linuxsecurity-installer` command.

   Note: You can use the `--override-distro` option to install the product on a distribution that is not officially supported. For example, `--override-distro rhel:8.6`. Note that WithSecure cannot offer support for any issues with unsupported distributions.

   Note: To use an HTTP proxy during the activation process, add the `--http-proxy` command line option. You can use this option in the following formats:

- `--http-proxy=host:port`: This configures the product to use the given `host` and `port` as the network proxy without any authentication. If no port number is given, port number 3128 is used by default. Example: `--http-proxy=proxy.example.com:8080`.
- `--http-proxy=username:password@host:port`: This configures the product to use the given `host` and `port` as the network proxy, and the given `username` and `password` as the credentials for authentication. Use URL encoding for any special characters in the username or password; for example, if the password contains an @ character, enter it as `%40`. Example: `--http-proxy=abc:x%40y%40z@proxy.example.com:8080`.

The following is a sample installation output when installing the product over the network (note that the first five lines of the sample output are Debian-specific):

```
Selecting previously unselected package f-secure-linuxsecurity.
(Reading database ... 26641 files and directories currently installed.)
Preparing to unpack .../f-secure-linuxsecurity.deb ...
Unpacking f-secure-linuxsecurity (12.0.9-1) ...
Setting up f-secure-linuxsecurity (12.0.9-1) ...

Installing F-Secure Linux Security...
Installing F-Secure BaseGuard...
Created symlink /etc/systemd/system/multi-user.target.wants/fsbg.service
/lib/systemd/system/fsbg.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/fsbg-statusd.service
/lib/systemd/system/fsbg-statusd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-lspmd.service

/lib/systemd/system/f-secure-linuxsecurity-lspmd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-statusd.service

/lib/systemd/system/f-secure-linuxsecurity-statusd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-webserver.service

/lib/systemd/system/f-secure-linuxsecurity-webserver.service.

The F-Secure license agreement is stored in:
  /opt/f-secure/linuxsecurity/doc/LICENSE
```

Tip: The installation has finished successfully as soon as the location of the license agreement is output.

For installation on isolated hosts using the content package, the command output includes additional lines. The following example is the output on a Red Hat Enterprise Linux 8.1 environment:

```
Preparing content for installation...
  linuxsecurity-1200-linux-x86_64.1585581846
  aqualnx64.1585581033
  fsbspamd-100-linux-x86_64.1582614359
  fmlibunix64.1580813614
  hydra-linux64.1585289384
  baseguard-100-linux-x86_64.1585583363
  fsbg-100-linux-x86_64.1585581843

Installing F-Secure Linux Security...
Installing F-Secure BaseGuard...
Created symlink /etc/systemd/system/multi-user.target.wants/fsbg-pmd.service
   /usr/lib/systemd/system/fsbg-pmd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/fsbg-statusd.service
/usr/lib/systemd/system/fsbg-statusd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-lspmd.service
   /usr/lib/systemd/system/f-secure-linuxsecurity-lspmd.service.
```

```
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-statusd.service
   /usr/lib/systemd/system/f-secure-linuxsecurity-statusd.service.
Created symlink
/etc/systemd/system/multi-user.target.wants/f-secure-linuxsecurity-webserver.service
   /usr/lib/systemd/system/f-secure-linuxsecurity-webserver.service.

The F-Secure license agreement is stored in:
  /opt/f-secure/linuxsecurity/doc/LICENSE
Installing virus definition databases...
  aqualnx64.1585581033
  fsbspamd-100-linux-x86_64.1582614359
  fmlibunix64.1580813614
  hydra-linux64.1585289384
```

In centrally managed installations, the host is shown in the **Pending hosts** list in Policy Manager Console after the installation is complete.

After you have installed the product, configure malware scanning and integrity checking settings to take them into use.

Related tasks

Using HTTP proxies on page 20
The product has a single, global HTTP proxy setting, which is always used when the product makes HTTP requests to external services.

2.3.1 Delaying product activation

Instead of activating the product immediately during installation or deployment, you can set the product activation to take place the next time the computer is turned on.

For example, you may want to delay activation if you are installing the product within a virtual machine template that is used for multiple virtual machine instances. In such a setup, you could install the product in the template and schedule activation for the next time the virtual machine is started. This approach allows individual virtual machines to be activated separately as they are taken into use.

1. To set activation to take place the next time the computer starts up, use the `--next-boot` argument in the installer command.

   For example:

   ```
   bash f-secure-linuxsecurity/f-secure-linuxsecurity-installer --next-boot
   ```

   This sets Linux Security 64 to be activated when the computer is next started.
2. The product activates automatically when the system is rebooted. No user interaction is necessary during this activation process.

2.4 Uninstalling the product

You can uninstall the product from the command line.

1. Log in to the Linux host as `root`.
2. Run the uninstallation command:

   - RHEL-based distributions: `rpm -e f-secure-linuxsecurity`
   - Debian-based distributions: `dpkg -r f-secure-linuxsecurity`

   If the above uninstallation command was not successful and the uninstallation fails, run the following command if Policy Manager Server is not installed on the same machine: `rm -rf /opt/f-secure /etc/opt/f-secure /var/opt/f-secure`. This command removes the remaining files or systemd services if any. Finally, restart the system to make sure that all hanging services and processes are run down.

   If you have installed Policy Manager Server on the same machine, do not delete the folders in full; instead, run the following commands to remove the necessary subfolders:

- `rm -rf /etc/opt/f-secure/baseguard`
- `rm -rf /etc/opt/f-secure/fsbg`
- `rm -rf /etc/opt/f-secure/linuxsecurity`
- `rm -rf /opt/f-secure/baseguard`
- `rm -rf /opt/f-secure/fsbg`
- `rm -rf /opt/f-secure/linuxsecurity`
- `rm -rf /var/opt/f-secure/baseguard`
- `rm -rf /var/opt/f-secure/fsbg`
- `rm -rf /var/opt/f-secure/linuxsecurity`

The uninstall command does not remove configuration files or log files that the product has generated. Check the product directories for remaining files and remove them if you are sure that you do not need them anymore.

Chapter

3

Using the product

Topics:

- Real-time scanning
- Integrity checking
- Automatic updates
- Using HTTP proxies
- Example: managing integrity checker profiles
- Basics of using F-Secure Policy Manager
- Configuring settings with the lsctl utility
- Command line usage

You can configure the product settings with the `lsctl` command-line tool or (for centrally managed computers) through F-Secure Policy Manager Console.

3.1 Real-time scanning

Real-time scanning protects the computer by scanning files when they are accessed and blocking access to files that contain ***malware***.

Note: Individual files and directories that require scanning in real time need to be specified in the Policy Manager Console. With the default settings, real-time scanning does not scan any files.

Real-time scanning works as follows:

1. The computer tries to access a file.
2. The file is immediately scanned for ***malware*** before the computer is allowed access to the file.
3. If ***malware*** is found in the file, real-time scanning blocks access to the file so the ***malware*** cannot harm the computer.
4. Based on the settings, real-time scanning may either rename or delete the infected file.

Scanned files are classified as clean, malware, potentially unwanted applications, or suspicious. Clean files are not affected by real-time scanning. Infected files are prevented from being opened.

The amount of time and system resources that real-time scanning takes depends on the contents, location, and type of the file.

Files that take a longer time to scan:

- Compressed files, such as .zip archives. Note that these files are not scanned by default.
- Files on network file systems.
- Large files may be affected.

Real-time scanning may slow down your computer when a lot of files are accessed at the same time.

Note: You can find all scan events in the `access.log` file.

Related Concepts

Command-line settings for real-time scanning on page 35
The settings related to real-time scanning that are available for `lsctl`.

Related Tasks

Using real-time scanning with Policy Manager on page 22
Real-time scanning protects the computer by scanning files when they are accessed and blocking access to files that contain ***malware***.

Scanning the computer manually from the command line on page 41
You can scan the computer for malware manually from the command line of a Linux host that has the product installed.

3.2 Integrity checking

Integrity checking protects important system files against unauthorized modifications.

You can use integrity checking to detect any modifications to protected files and prevent their use, regardless of file system permissions.

To use integrity checking, you need to add the files that you want to protect to a baseline list. The files on this list are protected against unauthorized changes.

Integrity checking works by comparing files on the disk to the baseline attributes, which form a cryptographically signed list of file properties. Integrity checking sends alerts to the administrator of attempts to modify the monitored files.

Note: Some aspects of integrity checking depend on the type of file system in use. Fully supported file systems include Ext4, ZFS, BTRFS, NFS, CiFS, and others. In essence, full integrity checking takes advantage of file attributes, which are available on most modern file systems, but not all.

The following table gives more details on how integrity checking handles various events for files that are included in the baseline.

Event	Integrity checking options
An untrusted process modifies a file	Configured with the read and write action settings: • If the write action is set to `deny`, the modification attempt is blocked and no alert is sent. • If the write action is set to `allow`, integrity checking sends an alert of the modification to the administrator. • If the read action is set to `deny`, integrity checking blocks any subsequent attempt to open the file after it has been modified.
A trusted process modifies a file	If the program file for a process is included in the baseline and has not been tampered with, it is considered a trusted process. In such cases, integrity checking does not take any action.
Deleting a file	Integrity checking does not prevent or send an alert for the event.
Renaming a file	Integrity checking does not prevent or send an alert for the event, but continues to monitor the renamed file.
Creating a hard or soft link to a file	Integrity checking follows the link.
Replacing a file	Integrity checking does not prevent or send an alert for the event.
Changing file permissions	The change is not blocked, but the read and write action settings determine if subsequent attempts to open the file are blocked or prompt an alert.
Changing file ownership	The change is not blocked, but the read and write action settings determine if subsequent attempts to open the file are blocked or prompt an alert.
Changing file attributes	This includes SELinux labels. Integrity checking does not prevent or send an alert for the event.

Note: Regarding the alerts that integrity checking sends:

- `tampering` alerts mean that a process is accessing a file that has already been tampered with.
- `tampering-action` alerts mean that a process is tampering with a file. These alerts are only sent if the write action is set to `allow`.
- If the write action is set to `allow`, all alerts are sent, regardless of the read action setting.

Related Concepts

Command-line settings for integrity checking on page 38
The settings related to integrity checking that are available for `lsctl`.

Related Tasks

Using integrity checking with Policy Manager on page 25
You can turn on integrity checking on the **Settings** > **Linux Security** page in Policy Manager Console.

Running the integrity checker from the command line on page 43
Integrity checking functionality is partially available on the command line.

3.3 Automatic updates

Automatic updates make sure that the Linux hosts in your managed network stay protected.

With automatic updates, the managed hosts retrieve the latest updates based on the policy settings. By default, automatic updates are turned on. We strongly recommend that you keep this feature turned on, as new virus definitions are made available several times a day.

F-Secure uses dedicated service connections between the backend systems and installed products, referred to as channels, to deliver updates to the malware definition databases, scanning engines, and the actual products. Each channel provides updates for a specific product component or function.

The settings for automatic updates apply to the three F-Secure product channels (`linuxsecurity`, `fsbg`, and `baseguard`) that the product uses. The automatic update settings do not affect updates for other channels, such as the scanning engines, as they are applied immediately when they are available. The exception to this is the main feature switch for automatic updates: turning automatic updates off disables the updates for all channels.

In addition, product channel updates are applied one week after the update has been published at the latest.

You can find information about the latest virus definition database update at https://dbtracker.f-secure.com/.

Related Concepts

Command-line settings for automatic updates on page 39
The settings related to automatic malware definition updates that are available for `lsctl`.

Related Tasks

Configuring automatic update options with Policy Manager on page 25
Configure automatic updates if you want to control how the latest updates are installed.

Updating the product manually using an update archive on page 44
In isolated environments with limited network connectivity, you can update the product and virus definition databases by creating an update archive and deploying it on the hosts.

3.4 Using HTTP proxies

The product has a single, global HTTP proxy setting, which is always used when the product makes HTTP requests to external services.

You can configure the global HTTP proxy setting using a command line option either when activating the product or with the `lsctl` utility after you have activated the product.

If you use the command line option to define the product settings, the product saves the configuration as a local setting override. This means that once you have finished activating the product, you can view the settings with the following command: `lsctl get http_proxy`.

Local overrides take precedence over the configuration that is retrieved from remote management services (F-Secure Elements Endpoint Protection or Policy Manager). To configure the proxy settings remotely, you have to mark the setting as locked in the management service. When the setting is locked, the value set in the profile or policy that is distributed from the management service is always used, even if a local configuration exists.

You can also change the proxy settings in the Policy Manager Console. The global HTTP proxy setting does not affect communication with Policy Manager.

When you use Policy Manager to manage the product, updates are downloaded from Policy Manager Server. The global HTTP proxy setting does not affect this communication either.

You can define one or more Policy Manager Proxies in Policy Manager Console. If you have defined a Policy Manager Proxy, this is used to handle the communication with Policy Manager. If you configure multiple Policy Manager Proxies, the product automatically switches to using the next available proxy if the connection with one proxy fails.

As the product must retrieve the Policy Manager Proxy configuration from Policy Manager Server, it must be able to make an initial connection to the main Policy Manager Server to get the proxy configuration.

If the product cannot connect to Policy Manager Server when trying to download updates, it falls back to using the F-Secure update servers directly. This communication does not use a HTTP proxy.

1. Set the HTTP proxy:
   - To set the proxy when you run the activation or installer command, include the `--http-proxy` option in the command:

     ```
     --http-proxy=<host address>:<host port number>
     ```

     Note: The product supports basic authentication for HTTP proxies. If the specified proxy requires authentication, use the `--http-proxy=username:password@host:port` format to specify the credentials for the proxy.
   - To set the proxy after activation, run the following `lsctl` commands:

     ```
     /opt/f-secure/linuxsecurity/bin/lsctl set http_proxy host <host address>
     /opt/f-secure/linuxsecurity/bin/lsctl set http_proxy port <host port
     number>
     /opt/f-secure/linuxsecurity/bin/lsctl set http_proxy enabled true
     ```

2. Run the following command to check the proxy setting:

   ```
   /opt/f-secure/linuxsecurity/bin/lsctl get http_proxy
   ```

   This returns the current settings:

   ```
   {
     "enabled": true,
     "host": <host address>,
     "port": <host port number>
   }
   ```

Related Tasks

Deploying the installation package on page 13
Once you have created the installation package, follow these instructions to deploy it on the target computers.

3.5 Example: managing integrity checker profiles

This example of a common task where you can use `lsctl` shows how to modify integrity checker profiles through the `ic profiles` array.

The following steps outline how to add an integrity checker profile to prevent writing to files inside the `/opt` directory.

1. Run the following command:

   ```
   /opt/f-secure/linuxsecurity/bin/lsctl set ic enabled yes
   /opt/f-secure/linuxsecurity/bin/lsctl add --file - ic profiles <<EOF
   ```

```
{
    "path": "/opt",
    "verify_attributes": {
        "mode": false,
        "user": false,
        "group": false,
        "size": false,
        "mtime": false
    },
    "read_action": "allow",
    "write_action": "deny"
}
EOF
```

This example uses shell's `here` document syntax for entering multi-line input.

2. To edit the profile, run the following command:

```
/opt/f-secure/linuxsecurity/bin/lsctl set --prompt ic profiles /opt
```

This opens a text editor with the current value of the `/opt` profile specified as a template.

3. To remove all integrity checker profiles, run the following command to set `ic profiles` to an empty array:

```
/opt/f-secure/linuxsecurity/bin/lsctl set ic profiles '[]'
```

3.6 Basics of using F-Secure Policy Manager

F-Secure Policy Manager Console is used to change the settings and to view statistics of F-Secure products.

Use the settings on the **Settings** > **Linux** pages to configure the product.

For more information about F-Secure Policy Manager, see the Policy Manager Admin Guide.

3.6.1 Using real-time scanning with Policy Manager

Real-time scanning protects the computer by scanning files when they are accessed and blocking access to files that contain ***malware***.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Real-time scanning** page.
3. Select **Enable real-time scanning**.
4. Enter the paths that you want to scan in the **Files and folders to scan** field.

   Use the full, absolute path names of individual files or directories. The listed directories also include subdirectories recursively. You can use wildcards in paths, with ** matching any subdirectories. Symbolic links are not followed.

   Note: By default, this field is empty, which means that nothing is scanned.

5. Enter any files and folders that you want to exclude from scanning.

   Exclusions also require the full, absolute path names for the files or directories. You can use wildcards in paths, with ** matching any subdirectories.

6. Select **Scan only executables** if you want to scan only files that have an executable permission flag turned on.
7. Under **Actions for real-time scanning**, set the **Action for malware** and **Action for suspicious files**.

   - **Rename**: Rename the infected file. The renamed file has a `.malware` or `.suspected` extension, depending on the type of detection.

- **Delete**: Remove the infected file.
- **Do nothing**: Do not perform any action on the infected file.

8. Click the following icon to distribute the policy:

Scanning for potentially unwanted applications

You can set both real-time and manual scanning to handle any detected potentially unwanted applications (PUA) in addition to malware.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Real-time scanning** page.
3. Make sure that **Enable real-time scanning** is selected.
4. Select **Scan for potentially unwanted applications**.

   Note: The PUA scanning settings are listed separately under both **Real-time scanning** and **Manual scanning**.

5. Under **Actions for real-time scanning** and **Actions for manual scanning**, set the **Action for potentially unwanted applications**.
   - **Rename**: Rename the infected file. The renamed file has a `.pua` extension.
   - **Delete**: Remove the infected file.
   - **Do nothing**: Do not perform any action on the infected file.
6. Click the following icon to distribute the policy:

Scanning archive files

You can set both real-time and manual scanning to check compressed archive files.

Archive scanning can scan files inside compressed `ZIP`, `ARJ`, `LZH`, `RAR`, `CAB`, `TAR`, `BZ2`, `GZ`, `JAR`, and `TGZ` archives.

Note: Archive scanning may need to uncompress the file content to disk temporarily. The space required for the temporary files depends on the content within the archive.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Real-time scanning** page.
3. Make sure that **Enable real-time scanning** is selected.
4. Under **Handling archives in real-time scanning**, select **Scan inside archives**.

   Note: The archive scanning settings are also listed separately under **Handling archives in manual scanning**.

5. If you want to automatically treat password-protected archive files as malware, select **Treat encrypted archives as unsafe**.
6. Set the maximum number of levels to scan within nested archives.

   Nested archives are archives inside other archives.
7. If you want to automatically treat archives that exceed the maximum number of nested levels as malware, select **Treat archives that exceed the maximum nesting level as unsafe**.

   Note: By default, archives that have more nesting levels than the set limit are treated as safe.

8. Click the following icon to distribute the policy:

Configuring manual scanning

You can set the exclusions and actions for manual scanning in Policy Manager.

While real-time scanning checks a file whenever it is opened or closed, manual scanning traverses all file systems methodically and looks for malicious files. You can exclude specific files and directories.

> **Note:** The settings for manual scanning are also used for scheduled scans.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Manual scanning** page.
3. Under **Manual scanning**, enter any files or folders that you want to exclude from scanning.

   Exclusions require the full, absolute path names for the files or directories. Wildcards are not supported.
4. Under **Actions for manual scanning**, set the **Action for malware** and **Action for suspicious files**.
   - **Rename**: Rename the infected file. The renamed file has a `.malware` or `.suspected` extension, depending on the type of detection.
   - **Delete**: Remove the infected file.
   - **Do nothing**: Do not perform any action on the infected file.
5. Click the following icon to distribute the policy:

> **Note:** Click the lock icon next to the settings if you do not want to allow users to override the policy settings.

Scanning a computer manually from Policy Manager

You can scan a managed Linux host manually from Policy Manager Console.

In the **Standard view** of Policy Manager Console:

1. Select the target host.
2. Select the **Operations** tab.
3. Click **Scan**.
   The manual scan starts on the selected host.
4. When the operation is complete, go to the **Scanning reports** tab to see the results.

Creating a scheduled scanning task

You can use scheduled scanning to scan the managed computers for ***malware*** at regular intervals.

> **Note:** Scheduled scanning uses the exclusions and actions set for manual scanning.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Manual scanning** page.
3. Under **Scheduled scanning**, enter the time that you want to run the scan.

   Use `*` to indicate any available option. For example, if you do not want to run the scan at a specific time, enter `*:**`.
4. Select the days of the week that you want to run the scan.

5. Click the following icon to distribute the policy:

3.6.2 Using integrity checking with Policy Manager

You can turn on integrity checking on the **Settings** > **Linux Security** page in Policy Manager Console.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Real-time scanning** page.
3. Under **Integrity checker**, select **Check the integrity of the following files**.
4. Check the protected files listed for the baseline.
5. Click the following icon to distribute the policy:

Adding files to the integrity checker baseline

You can add files to the integrity checker list to protect them against unwanted modifications.

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Real-time scanning** page.
3. Under **Integrity checker**, make sure that **Check the integrity of the following files** is selected.
4. Click **Add**.
5. Enter the path for the file that you want to protect.
6. Select the attributes that you want to monitor.

   - Mode: Changes to file permissions
   - User: Changes to file ownership
   - Group: Changes to file group
   - Size: Changes to file size
   - Modification time: Changes to file modification time

   The content of each listed file is always checked.
7. Select the read and write access permissions to the monitored file:

   - Select **File read** to prevent opening a tampered file.
   - Select **File write** to prevent any modifications to the protected file.
8. Click **Save**.
9. Click the following icon to distribute the policy:

3.6.3 Configuring automatic update options with Policy Manager

Configure automatic updates if you want to control how the latest updates are installed.

The settings for automatic updates apply to the three F-Secure product channels (`linuxsecurity`, `fsbg`, and `baseguard`) that the product uses. The automatic update settings do not affect updates for other channels, such as the scanning engines, as they are applied immediately when they are available. The exception to this is the main feature switch for automatic updates: turning automatic updates off disables the updates for all channels.

In addition, product channel updates are applied one week after the update has been published at the latest.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Centralized management** page.
3. Make sure that **Enable automated product and security updates** is selected.
4. Configure the scheduling for the automatic product updates:
   - **On arrival**: This setting is the default, and product and virus definition updates are applied as soon as they become available. When this is selected, **Date**, **Time**, and **Day** are ignored.
   - **Once**: Updates are postponed until the specified **Date** and **Time**. When this is selected, **Day** is ignored. The time is given in Policy Manager's local timezone.
   - **Daily**: Updates are applied at the specified **Time**. When this is selected, **Date** and **Day** are ignored.
   - **Weekly**: Updates are applied at the specified **Time** on the specified **Day**. When this is selected, **Date** is ignored.

   Note: Each update has an associated expiry time. Once the expiry time is reached, the update is applied regardless of the date and time setting. This means that users do not have to worry about late updates jeopardizing security.

5. If you want to keep using a specific version of the product, select **Use a specific product version instead of the latest version available** and enter the version that you want to use in the **Product version** field.

   Normally, all hosts are automatically updated to the latest available version, which means that there can be some changes to the product features without explicit notification. If you want to use a specific version, check the release notes for that version to find the text to enter here.
6. Select **Send alerts for failed updates** to generate product alerts for any failed updates.
7. Click the following icon to distribute the policy:

Note: If ***Linux Security 64*** cannot connect to ***Policy Manager*** to get the latest updates, it connects directly to the F-Secure update service.

Note: The **Operations** tab in Policy Manager Console includes an operation that you can use to control the virus definition and product updates manually.

Specifying an HTTP proxy

You can specify an HTTP proxy to use for features that require access to the internet.

Currently, only the reputation service (Object Reputation Service Platform or ORSP) depends on this setting, but other Security Cloud features could make use of it in the future as the product evolves.

Note: Policy Manager Server is assumed to not depend on this HTTP proxy setting but to be accessible directly. Furthermore, the software update service is provided by Policy Manager Server and is therefore not affected by the HTTP proxy setting.

In the **Standard view** of Policy Manager Console:

1. Select the target domain.
2. Go to the **Settings** > **Linux** > **Centralized management** page.
3. Select **Use HTTP proxy**.
4. Enter the host address and port number for the HTTP proxy.
5. Click the following icon to distribute the policy:

3.7 Configuring settings with the lsctl utility

You can check and edit the product settings using the `lsctl` command-line utility.

Note: If a setting for the distributed policy is locked in Policy Manager, that setting is enforced and you cannot edit it with the `lsctl` command-line utility.

To use the `lsctl` utility:

1. Log in to the Linux host as `root`.
2. Use the following commands:
    - To see the current value of a setting, run `/opt/f-secure/linuxsecurity/bin/lsctl get [OPTIONS] [SETTING]`
    - To assign a new value to a setting, run `/opt/f-secure/linuxsecurity/bin/lsctl set [OPTIONS] [SETTING] [VALUE]`
    - To add an entry to a setting that holds several values, run `/opt/f-secure/linuxsecurity/bin/lsctl add [OPTIONS] [SETTING] [VALUE]`
    - To remove an entry from a setting that holds several values, run `/opt/f-secure/linuxsecurity/bin/lsctl del [OPTIONS] [SETTING] [KEY]`
    - To clear the value of a setting, run `/opt/f-secure/linuxsecurity/bin/lsctl reset [OPTION] [SETTING]`
    - To see a list of the operations available for a setting, run `/opt/f-secure/linuxsecurity/bin/lsctl help [OPTIONS] [SETTING]`
    - To load product settings from a backup, run `/opt/f-secure/linuxsecurity/bin/lsctl load [OPTIONS] [SETTING] [VALUE]`
    - To get a list of all available settings, run `/opt/f-secure/linuxsecurity/bin/lsctl -h`

The `get`, `set`, `add`, `del`, and `load` commands support the following options:

--json, -j Treat input as JSON and produce output in JSON format.

--raw, -r This is currently the default. Raw is an alternative input/output format that is similar to JSON with the following changes:

- String values are accepted and returned without surrounding quotes.
- Boolean values accept additional expressions such as `yes` and `no` in addition to `true` and `false`.

Note: The default input/output format for `lsctl` is subject to change. If you rely on the format of `lsctl`, specify either `--json` or `--raw` explicitly when `lsctl` is run. In addition to setting values, the keys for array-type settings are considered as input and need to be entered in the correct format.

The `set`, `add`, and `load` commands accept the following options:

--prompt, -p Prompt for input using a text editor.

--file, -f [PATH] Read the input value from a file. If the path is set to `-`, the input value is read from the standard input.

If you use the `--prompt` or `--file` options, do not enter a `[VALUE]` for the command.

Note: The `set` and `load` commands differ in how they handle settings that are not included in the input. The `set` command does not change the current values for any settings that are missing from the input, whereas the `load` command resets those settings to their default values.

3.7.1 Setting types and the settings tree

Each product setting has a type and a value, and the settings are organized in a tree-like structure.

There are five basic types of settings:

- Boolean (`true` or `false`)
- Text
- Numbers
- Objects, which combine a specific group of settings in one entity by assigning a name to each member. You can inspect and assign values to objects.
- Arrays, which are collections that contain a varying number of members. Each array member is identified by a unique key. Array keys are composed of the same basic value types as settings, for example a key might be a string, number, or object. You can add and delete the settings in an array. Each array can have additional restrictions on the type of settings that it accepts, for example it may only accept text-type settings as members.

Product settings are organized in a tree structure, where the root is an object-type setting that contains all the other settings. To inspect this root setting, use the `get` sub-command, for example:

```
# /opt/f-secure/linuxsecurity/bin/lsctl get
```

```
{
  "http_proxy": {
    "enabled": false,
    "host": "localhost",
    "port": 3128
  },
  "ic": {
    "enabled": true,
    "profiles": []
  },
  "oas": {
    "actions": {
      "malware": "rename",
      "pua": "none",
      "suspected": "none"
    },
    "archive_max_nested": 5,
    "block_archive_max_nested": false,
    "block_encrypted_archives": false,
    "detect_pua": true,
    "enabled": true,
    "exclude_paths": [],
    "include_paths": [],
    "scan_archives": false,
    "scan_only_executables": false
  },
  ...
}
```

Note that this example does not include the full output.

The output shows the setting hierarchy, starting from the root object and continuing through all its child settings (`http_proxy`, `ic`, `oas`, etc.) and their sub-settings recursively.

In this example, the root object contains the `http_proxy` setting, which is an object-type setting that contains three sub-settings: `enabled` (boolean value setting), `host` (text setting), and `port` (number setting).

You can also use the `get` sub-command to inspect only parts of the setting tree. It selects the settings to retrieve using the names that object-type settings assign and the keys of array-type settings.

For example, you can retrieve the HTTP proxy settings alone as follows:

```
# /opt/f-secure/linuxsecurity/bin/lsctl get http_proxy
```

```
{
  "enabled": false,
  "host": "localhost",
  "port": 3128
}
```

In the same way, you can specify the command further to check if the use of an HTTP proxy is enabled by running:

```
# /opt/f-secure/linuxsecurity/bin/lsctl get http_proxy enabled
```

```
false
```

In addition to their type, settings can have additional constraints on the allowed values. For example, the `http_proxy port` setting does not allow negative numbers.

3.7.2 Operating with settings

There are six basic sub-commands for operating with settings: `get`, `set`, `add`, `delete`, `reset`, and `load`.

get `lsctl get [--json | --raw] SETTING...`

Use the `get` sub-command to inspect the current value of a setting.

set `lsctl set [--json | --raw] [--prompt | --file PATH] SETTING... VALUE`

Use the `set` sub-command to enter the value for a setting.

add `lsctl add [--json | --raw] [--prompt | --file PATH] SETTING... VALUE`

Use the `add` sub-command to add new members to an array-type setting. The key that is used for the new array member is automatically derived from the value. How the key is derived is specific to each particular setting.

delete `lsctl delete [--json | --raw] SETTING... KEY`

Use the `delete` sub-command to remove members from array-type settings. This means that you can only use it with settings that are immediate descendants of an array-type setting. You can use `del` as an alias for the `delete` sub-command.

reset `lsctl reset [--json | --raw] [--all] SETTING...`

Use the `reset` sub-command to clear the local value of a setting. This means that the setting's value is set back to its default value. This also enables the setting to be changed using Policy Manager or Protection Service for Business.

load `lsctl load [--json | --raw] [--prompt | --file PATH] SETTING... VALUE`

Use the `load` sub-command to restore the values for settings.

Note: The `set` and `load` commands differ in how they handle settings that are not included in the input. The `set` command does not change the current values for any settings that are missing from the input, whereas the `load` command resets those settings to their default values.

Providing values for settings

The `set` and `add` sub-commands are used to alter setting values. Use the final command-line argument to enter the value to assign to a setting (for `set` sub-commands) or the member to add to an array-type setting (for `add` sub-commands).

For example, to set a new value for the `http_proxy host` setting, you could use the following command:

```
lsctl set http_proxy host example.com
```

This assigns `example.com` as the new value for the `host` text-type setting.

When you use the `--prompt` (`-p`) flag, `lsctl` opens a text editor for you to specify the value for a setting. For example:

```
lsctl set --prompt http_proxy host
```

This opens a text editor that you can use to enter a new value for the `host` setting.

By default, `lsctl` opens the editor specified by the `EDITOR` environment variable. If `EDITOR` is not set or the specified editor is not found in `PATH`, `lsctl` tries to find the following editors in order until it finds one that is available: `nano`, `vim`, `vi`.

When you use the `--file` (`-f`) switch, `lsctl` reads the value for a setting from a file. For example:

```
lsctl set --file value.txt http_proxy host
```

This reads the value for the `host` setting from the `value.txt` file. You can also use the special - value to read the value from the standard input. For example:

```
echo -n 'example.com' | lsctl set --file - value.txt http_proxy host
```

This uses the output of the `echo` command as the value for the `host` setting.

Related Tasks

Loading product settings from a file on page 32
You can use the `lsctl load` command to restore settings from a previously saved JSON file.

3.7.3 Input and output formats

The format defines how the setting values that you provide are interpreted and how `lsctl` prints the setting values in its output.

An input format defines how the entered setting values should be formatted. The input format affects all sources of inputs. This means that it does not matter if the setting value is entered using command line arguments or if it is read from a file (by using the `--file` switch); all input must follow the same formatting rules.

An output format defines how setting values are formatted when `lsctl` prints them.

`lsctl` currently supports two formats: `json` and `raw`. You can select these using the `--json` (`-j`) and `--raw` (`-r`) flags respectively. By default, `lsctl` currently uses the `raw` format, but this is subject to change. If you depend on the exact input or output format of `lsctl`, always specify the desired format explicitly by using these flags.

JSON

When you specify the `--json` flag, `lsctl` expects the input to be valid JSON and outputs valid JSON. For example:

```
lsctl set --json http_proxy host '"example.com"'
```

Note the need for double quotes. JSON strings are always enclosed in double quotes (`"`). However, because of the way `shell` handles word splitting, you need to enclose the JSON string within an additional pair of single quotes to prevent `shell` from removing the quotes from the value. As this is something specific to the way `shell` handles command line arguments, you do not need this when the value is read from a file (by using the `--file` switch) or received from an interactive editor (the `--prompt` flag).

Format flags also affect the output format. For example:

```
lsctl get --json http_proxy host
```

```
"example.com"
```

Note that the output is a valid JSON string literal.

Raw

When you specify the `--raw` flag, `lsctl` uses the `raw` format. The `raw` format modifies the JSON format slightly to make certain operations more ergonomic for command line users. The differences between `json` and `raw` formats are:

- Text values for settings do not need to be enclosed in double quotes. This means that the following command is valid:

  ```
  lsctl set http_proxy host example.com
  ```

- Boolean values for settings support additional literals (`yes`, `y`, `no` and `n`) in addition to `true` and `false`. This makes the following command valid:

  ```
  lsctl set http_proxy enabled yes
  ```

Note that these additions only apply when you are operating directly with text or boolean value settings. They are not applied when these settings appear as a part of other settings. Therefore, the following command **is not valid**:

```
lsctl set http proxy '{"enabled": yes, "host": example.com, "port": 9090}'
```

3.7.4 Working with arrays

Arrays differ from other settings in that they can contain a variable number of member settings.

You can add members to and remove them from arrays using the `add` and `delete` sub-commands.

For example, you can add a new directory to include in on-access scanning by adding a new member to the `oas include_paths` setting, which is an array setting that contains string members:

```
lsctl add oas include_paths /home
```

The above command adds the `/home` directory to the array.

To remove the newly added directory from `include_paths`, use the following command:

```
lsctl del oas include_paths /home
```

Arrays can also include more complex settings are also possible. For example, integrity checker profiles are stored in the `ic profiles` array. Each integrity checker profile is an object that defines the profile properties. To add a new integrity checker profile, you could use the following command:

```
lsctl add --file - ic profiles <<EOF
{
    "path": "/home",
    "verify_attributes": {
        "mode": false,
        "user": false,
        "group": false,
        "size": false,
        "mtime": false
    },
    "read_action": "deny",
    "write_action": "allow"
}
EOF
```

Note: This example uses shell's `here` document syntax to make it easier to enter multi-line values.

Note the use of the `--file` switch with the `-` argument to indicate that the value should be read from the standard input. Alternatively, you could use the `--prompt` flag to make entering large input values easier.

To delete the newly added profile, use the following command:

```
lsctl del ic profiles /home
```

Here the `path` member is the key that is used to identify members within the `profiles` array. How an array's elements are identified is specific to each array.

You can also access settings within arrays using the usual sub-commands. For example, you could retrieve the value of the `write_action` setting inside the newly added profile:

```
lsctl get ic profiles /home write_action
```

```
allow
```

Or you could change the value of the `mode` setting within `verify_attributes`:

```
lsctl set ic profiles /home verify_attributes mode yes
```

In addition to affecting the way that setting values are handled, input and output formats also affect how you enter array keys. For example, if you want to retrieve the value of `write_action` settings when using the JSON format, you would have to enter the `lsctl` command like this:

```
lsctl get --json ic profiles '"/home"' write_action
```

```
"allow"
```

3.7.5 Loading product settings from a file

You can use the `lsctl load` command to restore settings from a previously saved JSON file.

This allows you to save your tested, functioning product configuration as a backup and then restore it if necessary. To save the current product configuration, you could use the following command, for example:

```
lsctl get --json > [PATH]
```

For this command, replace `[PATH]` with the full path and filename for the JSON backup file.

To restore the product settings, run the following command:

```
lsctl load --json --file [PATH]
```

This applies all the settings that are available in the backup file specified as `[PATH]` to your product configuration. Any settings that are missing are reset to their default value.

3.7.6 Using lsctl and Policy Manager

By default, when you use `lsctl` to change settings locally, those local settings override the centrally managed setting values received from Policy Manager.

This behavior changes if a setting is marked as locked in Policy Manager. If a setting is locked, the value that is set for it in Policy Manager is always used.

You can use the `reset` sub-command to remove a local override and revert the setting back under the control of Policy Manager. For example, you can use `lsctl` to disable on-access scanning locally:

```
lsctl set oas enabled false
```

Now the on-access scanning status has been set locally and can no longer be controlled using Policy Manager, except by locking the setting. If the setting is changed using Policy Manger, those changes are not taken into use.

To remove the local setting override and allow Policy Manager to control the on-access scanning status, use the `reset` sub-command:

```
lsctl reset oas enabled
```

Now the setting can again be changed using Policy Manager.

3.7.7 Inspecting the state of the product

You can use the `lsctl` command-line utility to inspect the state of the product and to view various scanning and performance statistics with the `status` sub-command.

Inspecting different status values works similarly to inspecting setting values:

- To get a status value, run `lsctl status get [--json | --raw] STATUS...`
- To see the help for a status, run `lsctl status help [--json | --raw] STATUS...`

scan_service	Type: JSON object All statistics related to the scanning service.
scan_service connection_count	Type: integer The number of connected hosts.
scan_service last_detection	Type: string The name of the last detection.
scan_service max_connection_count	Type: JSON object All statistics related to the maximum number of connected hosts.
scan_service max_connection_count daily	Type: integer The maximum number of connected hosts in the last 24 hours.
scan_service max_connection_count weekly	Type: integer The maximum number of connected hosts in the last week.
scan_service max_connection_count monthly	Type: integer The maximum number of connected hosts in the last month.
scan_service transaction_count	Type: JSON object All statistics related to the scanning transaction figures.
scan_service transaction_count total	Type: integer The total number of transactions.
scan_service transaction_count daily	Type: integer The number of transactions in the last 24 hours.
scan_service transaction_count weekly	Type: integer The number of transactions in the last week.
scan_service transaction_count monthly	Type: integer The number of transactions in the last month.
scan_service detection_count	Type: JSON object All statistics related to the detection figures.
scan_service detection_count total	Type: integer The number of detections.
scan_service detection_count daily	Type: integer The number of detections in the last 24 hours.
scan_service detection_count weekly	Type: integer

The number of detections in the last week.

scan_service detection_count monthly

Type: integer

The number of detections in the last month.

scan_service min_transaction_latency

Type: JSON object

All statistics related to the minimum latency in scanning transactions.

scan_service min_transaction_latency daily

Type: integer

The minimum transaction latency (milliseconds) in the last 24 hours.

scan_service min_transaction_latency weekly

Type: integer

The minimum transaction latency (milliseconds) in the last week.

scan_service min_transaction_latency monthly

Type: integer

The minimum transaction latency (milliseconds) in the last month.

scan_service max_transaction_latency

Type: JSON object

All statistics related to the maximum latency in scanning transactions.

scan_service max_transaction_latency daily

Type: integer

The maximum transaction latency (milliseconds) in the last 24 hours.

scan_service max_transaction_latency weekly

Type: integer

The maximum transaction latency (milliseconds) in the last week.

scan_service max_transaction_latency monthly

Type: integer

The maximum transaction latency (milliseconds) in the last month.

scan_service avg_transaction_latency

Type: JSON object

All statistics related to the average latency in scanning transactions.

scan_service avg_transaction_latency daily

Type: integer

The average transaction latency (milliseconds) in the last 24 hours.

scan_service avg_transaction_latency weekly

Type: integer

The average transaction latency (milliseconds) in the last week.

scan_service avg_transaction_latency monthly

Type: integer

The average transaction latency (milliseconds) in the last month.

product_version

Type: JSON object

All information related to the product version.

product_version name

Type: string

The name of the pinned product version. This can be an empty string if no pinned product version has been specified.

Note: If you use a pinned version of the product, it works for one year from the date when the new pinnable product version is released. You should update the product to the new version during that time.

product_version status

Type: string

A string representing the status of the pinned product version. The value can be `unset`, `error`, `in-progress`, or `applied`.

3.7.8 Command-line settings for real-time scanning

The settings related to real-time scanning that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

oas

Type: JSON object

All settings related to real-time scanning.

oas enabled

Type: boolean

Controls whether real-time scanning is on or off.

oas include_paths

Type: JSON array

The paths included in real-time scanning.

oas include_paths [PATH]

Type: string

A path included in real-time scanning. For use with `add` and `del` commands. `[PATH]` must be a full, absolute path name.

You can use wildcards in paths, with ** matching any subdirectories.

oas exclude_paths

Type: JSON array

The paths excluded from real-time scanning.

oas exclude_paths [PATH]

Type: string

A path excluded from real-time scanning. For use with `add` and `del` commands. `[PATH]` must be a full, absolute path name.

You can use wildcards in paths, with ** matching any subdirectories.

oas actions

Type: JSON object

The actions configuration for real-time scanning.

oas actions malware

Type: string

The action on detecting files that are identified as malware. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.malware` extension to the file name
- `none` - do not do anything to the file

oas actions suspected

Type: string

The action on detecting files that are suspected to be unsafe. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.suspected` extension to the file name
- `none` - do not do anything to the file

oas detect_pua

Type: boolean

Controls whether scanning for potentially unwanted applications (PUA) is on or off.

oas actions pua

Type: string

The action on detecting files that are identified as potentially unwanted applications. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.pua` extension to the file name
- `none` - do not do anything to the file

oas scan_archives

Type: boolean

Controls whether scanning inside archive files is on or off.

oas block_archive_max_nested

Type: boolean

Controls whether or not archive files that are too deeply nested are considered unsafe.

oas archive_max_nested

Type: integer

The maximum allowed level of nesting for archive files. If `oas block_archive_max_nested` is set to `true`, archive files that exceed this level of nesting are considered unsafe.

oas block_encrypted_archives

Type: boolean

Controls whether or not encrypted archive files are considered unsafe.

oas scan_only_executables

Type: boolean

Controls whether or not scanning is restricted to files that have execute permissions.

Settings for manual and scheduled scanning

The settings related to manual and scheduled scanning that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

ods

Type: JSON object

All settings related to manual and scheduled scanning.

ods exclude_paths

Type: JSON array

The paths excluded from scheduled scanning.

ods exclude_paths [PATH]

Type: string

A path excluded from scheduled scanning. For use with `add` and `del` commands. `[PATH]` must be a full, absolute path name.

You can use wildcards in paths, with ** matching any subdirectories.

ods actions

Type: JSON object

The actions configuration for manual and scheduled scanning.

ods actions malware

Type: string

The action on detecting files that are identified as malware. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.malware` extension to the file name
- `none` - do not do anything to the file

ods actions suspected

Type: string

The action on detecting files that are suspected to be unsafe. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.suspected` extension to the file name
- `none` - do not do anything to the file

ods detect_pua

Type: boolean

Controls whether scanning for potentially unwanted applications (PUA) is on or off.

ods actions pua

Type: string

The action on detecting files that are identified as potentially unwanted applications. Allowed values are:

- `remove` - delete the file
- `rename` - add a `.pua` extension to the file name
- `none` - do not do anything to the file

ods scan_archives

Type: boolean

Controls whether scanning inside archive files is on or off.

ods block_archive_max_nested

Type: boolean

Controls whether or not archive files that are too deeply nested are considered unsafe.

ods archive_max_nested

Type: integer

The maximum allowed level of nesting for archive files. If `ods block_archive_max_nested` is set to `true`, archive files that exceed this level of nesting are considered unsafe.

ods block_encrypted_archives

Type: boolean

Controls whether or not encrypted archive files are considered unsafe.

ods schedule

Type: JSON object

All settings related to scheduled scanning.

ods schedule schedule_type

Type: string

Type of scheduled scanning in use. Currently the only allowed value is `weekly`.

ods schedule weekly_schedule

Type: JSON object

Settings for scanning on a weekly schedule.

ods schedule weekly_schedule [monday|tuesday|wednesday|thursday|friday|saturday|sunday]

Type: boolean

This group of settings controls if the scheduled scan is run on the given weekday.

ods schedule weekly_schedule time_of_day

Type: string

The time to start a scheduled scan on the set days. The value is a 24-hour time value (`HH:MM`) in the endpoint's time zone.

ORSP settings

The settings related to the Object Reputation Services Platform (ORSP) that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

orsp

Type: JSON object

Settings related to using the cloud-based Object Reputation Services Platform (ORSP) in scanning.

orsp enabled

Type: boolean

Controls whether ORSP is switched on or off for file scanning.

3.7.9 Command-line settings for integrity checking

The settings related to integrity checking that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

ic

Type: JSON object

All settings related to integrity checking.

ic enabled

Type: boolean

Controls whether integrity checking is on or off.

ic profiles

Type: JSON array

The baseline list of path names included in integrity checking.

ic profiles [PATH]

Type: JSON object

All integrity checking settings for the given path.

ic profiles [PATH] path

Type: string

A path name included in integrity checking.

ic profiles [PATH] verify_attributes

Type: JSON object

All monitored attributes for the given path.

ic profiles [PATH] verify_attributes [mode|user|group|size|mtime]

Type: boolean

This group of settings controls whether to monitor changes in the file mode, the owning user or group, size, or modification time of the given path.

ic profiles [PATH] read_action

Type: string

The action on an attempt to read from the given path if its contents or monitored attributes have changed. The allowed values are `allow` and `deny`, which respectively permit or prevent access to the file.

ic profiles [PATH] write_action

Type: string

The action on an attempt to write to the given path. The allowed values are `allow` and `deny`, which respectively permit or prevent access to the file.

ic profiles [PATH] exclude_patterns

Type: JSON array

List of file name patterns to be excluded from integrity checking. For example, "*.log" pattern could be used to exclude all files with .log file extension from integrity checking.

Note: You can use wildcards in file name patterns, but the path does not support them.

ic system_updates

Type: boolean

This group of settings controls whether baseline is automatically updated whenever packages are installed or upgraded with package managers.

ic system_updates apt

Type: boolean

This group of settings is for the APT package management system.

ic system_updates apt enabled

Type: boolean

When the apt integration is enabled, integrity checking updates the baseline automatically whenever packages are installed or upgraded with `apt` and its related tools like `apt-get`.

ic system_updates dnf

Type: boolean

This group of settings is for the DNF package manager.

ic system_updates dnf enabled

Type: boolean

When the dnf integration is enabled, integrity checking updates the baseline automatically whenever packages are installed or upgraded with the `dnf` package manager.

3.7.10 Command-line settings for offload scanning

The settings related to offload scanning that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

offload_scanning

Type: boolean

All settings related to offload scanning.

offload_scanning enabled

Type: boolean

Controls whether offload scanning is on or off.

offload_scanning icap_servers

Type: JSON array

List of ICAP servers that can be used for offloading scans.

offload_scanning icap_servers [SERVER]

Type: string

The ICAP server settings.

offload_scanning icap_servers [SERVER] address

Type: string

The address of the ICAP server that performs the offload scanning.

offload_scanning icap_servers [SERVER] port

Type: integer

The port number for offload scanning on the ICAP server.

offload_scanning timeout

Type: integer

This setting controls the time, in seconds, to wait for a response from the offloading server.

3.7.11 Command-line settings for automatic updates

The settings related to automatic malware definition updates that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

updates

Type: JSON object

All settings related to automatic updates.

updates enabled

Type: boolean

Controls whether automatic updates are switched on or off.

updates product_version

Type: string

If not empty, this sets a specific product version to use instead of automatically updating to the latest available version.

updates schedule

Type: JSON object

The automatic update schedule.

updates schedule regime

Type: string

The type of scheduling for automatic updates. The allowed values are:

- `on_arrival` - updates are applied as soon as they become available
- `at_date` - updates are postponed until the time specified in the `at_date_schedule` setting
- `with_repetition` - updates are applied regularly according to the `repetition_schedule` settings

updates schedule at_date_schedule

Type: integer

The time to apply updates when `regime` is set to `at_date`. The value must be a timestamp in epoch time format.

updates schedule repetition_schedule

Type: JSON object

Settings related to automatic updates with a fixed repetition schedule.

updates schedule repetition_schedule day

Type: string

The scheduled day for repeated updates. Allowed values are `daily`, to check for updates every day at the specific time, or `monday`, `tuesday`, `wednesday`, `thursday`, `friday`, `saturday`, or `sunday` to check for updates on the given day each week.

updates schedule repetition_schedule time_of_day

Type: string

The scheduled time of day for applying updates. The value is a 24-hour time value (`HH:MM`).

updates use_tls

Type: boolean

Controls whether Transport Layer Security (TLS) is used when downloading updates.

Related Concepts

Services installed with the product on page 49
The product installs several services that are used to handle various product features.

Related Tasks

Updating the product manually using an update archive on page 44
In isolated environments with limited network connectivity, you can update the product and virus definition databases by creating an update archive and deploying it on the hosts.

HTTP proxy settings

The settings related to HTTP proxy usage that are available for `lsctl`.

Tip: To see an example of the structure for any of the settings, you can check the output for the following command: `/opt/f-secure/linuxsecurity/bin/lsctl get [--json|--raw] [SETTING]`

http_proxy

Type: JSON object

All HTTP proxy settings.

http_proxy enabled

Type: boolean

Controls whether or not to use an HTTP proxy for features that require network access.

http_proxy host

Type: string

The host name of the HTTP proxy.

http_proxy port

Type: integer

The HTTP proxy port. The value must be a number between `1` and `65535`.

3.8 Command line usage

This section describes the commands that you can run from the command line in the product.

3.8.1 Starting and stopping services

The product contains a number of background services, which you can control using the `/opt/f-secure/fsbg/bin/master-switch` command.

You can use the `master-switch` command to control all the services related to the product. The syntax for the command is:

```
/opt/f-secure/fsbg/bin/master-switch ( on | off | status | enable | disable
| is-enabled )
```

Note: Policy Manager controls whether or not the `master-switch` command can be used for turning off F-Secure services. Change this using the **Linux** > **Centralized management** > **Allow users to unload products** setting.

Use the `off` sub-command to temporarily stop all F-Secure services:

```
/opt/f-secure/fsbg/bin/master-switch off
```

The on sub-command restarts the stopped services:

```
/opt/f-secure/fsbg/bin/master-switch on
```

To view the status of F-Secure services, use the `status` sub-command:

```
/opt/f-secure/fsbg/bin/master-switch status
```

The `disable` sub-command turns off all F-Secure services. The services stay turned off after you restart the computer. The `enable` sub-command turns on all F-Secure services again. Use the `is-enabled` sub-command to check if F-Secure services are turned on.

3.8.2 Scanning the computer manually from the command line

You can scan the computer for malware manually from the command line of a Linux host that has the product installed.

The `fsanalyze` program scans files with the privileges of the user who runs it. The user must have read access to files, and read and execute access to all directories to be scanned. To rename or remove infected files, the user must have write access to the directories where the files are located.

The `fsanalyze` command scans specified targets (files or directories) and reports any malicious code it detects.

If no files are specified on the command line, `fsanalyze` reads the standard input for content to analyze. Otherwise, every given file is analyzed and every given directory is scanned recursively, with the following exceptions:

- Special files (such as socket, FIFO, or device files), and files on the `/proc` and `/sys` special file systems are skipped.
- Recursive scanning does not follow symbolic links unless this is explicitly requested with the `--follow` command line option.
- Recursive scanning does not automatically cross file system boundaries. To scan files mounted on other file systems, specify the mount points explicitly on the command line.

Note: The exclusions that are set in the current policy for manual scanning are applied when you use manual scanning from the command line. If you have excluded any paths in the manual scanning policy, recursive scanning crosses file system boundaries.

Run the following command from the shell to start the manual scan:

```
/opt/f-secure/linuxsecurity/bin/fsanalyze [OPTIONS] [FILE...]
```

Usage: `fsanalyze [OPTIONS] [FILE...]`

Analyze each `FILE` for potential malicious content. If no file is specified, analyze standard input, in which case the only supported action is `none`.

-h, --help
Show help.

--malware=ACTION
Action to take when a file is detected as malware. If ACTION is `remove` the file is removed; if ACTION is `rename` the file is renamed adding the detection type as an extension; if ACTION is `none`, the infection is only reported on standard output.

Default: `rename`

--pua=ACTION
Action to take when a file is detected as a potentially unwanted application. Options for the action are the same as for malware.

Default: `none`

--suspected=ACTION
Action to take when a file is detected as suspicious. Options for the action are the same as for malware.

Default: `none`

-L, --follow
Follow symbolic links.

-l, --list
List all scanned files.

--preserve-atime=VALUE
Preserve the access time of files that are scanned. Value can be `yes` or `no`.

Default: `no`

-q, --quiet
Only output the scan results.

--scan-archives=VALUE
Enable archive scanning. Value can be `yes` or `no`.

Default: `no`

--max-nested=NUMBER
Set the maximum allowed nesting level of scanned archives.

Default: 5

--detect-max-nested=VALUE
If enabled, treat archives that exceed the maximum depth as malware. Value can be `yes` or `no`.

Default: `no`

--detect-encrypted-archives=VALUE
Treat encrypted archives as malware. Value can be `yes` or `no`.

Default: `no`

--detect-pua=VALUE
If disabled, potentially unwanted applications are ignored. Value can be `yes` or `no`.

Default: `yes`

--use-orsp=VALUE
If enabled, use the reputation service (ORSP) from Security Cloud. Value can be `yes` or `no`.

Default: `yes`

-v, --version
Display program version and exit.

--quoting-style=STYLE
Set how to quote the output. If STYLE is `url`, the output is URL-encoded; if STYLE is `escape`, ASCII control characters in the output are escaped as `<NAME>`, and if the output is not

	redirected to a file, they are also highlighted; if STYLE is `literal`, the output is printed as is. Default: `url`
--ignore-exclude-paths	Does not use the exclusions set from the current policy for manual scanning.

The scanning exit codes are as follows:

- `0` = scanning was completed with no errors, and no malware, PUA, or suspicious content was detected
- `1` = scanning failed for one or more files
- `2` = scanning was completed with no errors, but malware, PUA, or suspicious content was detected

3.8.3 Testing the antivirus protection

To test whether the product operates correctly, you can use a special test file that is detected as a virus.

The EICAR (EICAR is the European Institute of Computer Anti-virus Research) standard antivirus test file is detected by several antivirus programs. The Eicar info page can be found at https://www.f-secure.com/v-descs/eicar.shtml.

1. Download or create the EICAR test file.
   - Download the EICAR test file from https://www.f-secure.com/v-descs/eicar.shtml, or
   - Use any text editor to create the eicar.com file with the following single line in it: `X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*`
2. Run the following command: `/opt/f-secure/linuxsecurity/bin/fsanalyze eicar.com`

The product should detect the EICAR test file as a virus.

3.8.4 Running the integrity checker from the command line

Integrity checking functionality is partially available on the command line.

To run the integrity checker from the command line on a Linux host:

1. Log in to the Linux host as `root`.
2. Run the following command from the shell: `/opt/f-secure/linuxsecurity/bin/fsic`

   Usage:
   - `fsic verify`: This checks each file in the baseline that is configured for the policy. It checks if the file has been tampered with and also scans the file for malware. The command prints every file that fails verification to the standard output.
   - `fsic unprotect`: This allows baseline files to be modified.
   - `fsic protect`: This regenerates the baseline index and prevents modifications to the baseline files.
3. Check the exit code for the command:
   - `0` = No errors or failed verification results were encountered
   - `1` = An error occurred that prevented the command from running successfully
   - `2` = One or more files have been tampered with or contained malware, PUA, or suspicious content

3.8.5 Updating the product manually from the command line

You can update the product, the malware definition databases, and the scanning engines manually to their latest versions using the `lsctl` program.

Note: The instructions in this topic do not apply to isolated product installations.

You can update the product from the command line by running the following command as the root user:

```
/opt/f-secure/linuxsecurity/bin/lsctl update
```

This command checks for the availability of new updates over the network, and downloads and installs them on the system to make sure that everything is up to date.

The `--timeout` option can be used to specify the period of time to wait, in seconds, for the update to complete before letting it finish in the background. The value `0`, which is used as the default if the option is not specified, disables the timeout.

Note: There are a few things to consider before and during the update:

- Before running the command, make sure that the product services are running.
- Once the update has started, downloading and installing the updates cannot be canceled. When using the `--timeout` option, the operation is left to complete in the background after the timeout.
- If a specific product version has been set in the configuration, the product is not updated past that version. Only the malware definitions and the scanning engine updates are installed.

Related Concepts

Starting and stopping services on page 41
The product contains a number of background services, which you can control using the `/opt/f-secure/fsbg/bin/master-switch` command.

Related Tasks

Updating the product manually using an update archive on page 44
In isolated environments with limited network connectivity, you can update the product and virus definition databases by creating an update archive and deploying it on the hosts.

3.8.6 Updating the product manually using an update archive

In isolated environments with limited network connectivity, you can update the product and virus definition databases by creating an update archive and deploying it on the hosts.

Note: To generate the update archive, use a computer that has network access to F-Secure servers and either Policy Manager Server (Linux or Windows) installed or a separately available `fspm-definitions-update-tool` downloaded (Linux).

Follow these instructions to create the update archive and install the updates with the update tool.

Note: These instructions are for Linux. You can find the command path and the syntax for Windows in the Policy Manager administrator's guide.

1. Open a command line.

   This computer must have a network connection so that it can connect to F-Secure servers.
2. Select the directory where you want to extract the Policy Manager definitions update tool.

   Make sure that you use an existing directory. The examples in these steps use `<DIR>` as a placeholder for this directory.
3. Extract the definitions update tool.
   - If you have Policy Manager Server, run the following command:

     ```
     /opt/f-secure/fspms/bin/prepare-fspm-definitions-update-tool <DIR>
     ```
   - If you have downloaded the fspm-definitions-update-tool.tar.gz file, run the following command:

     ```
     tar -C <DIR> -xf /PATH/TO/fspm-definitions-update-tool.tar.gz
     ```
4. Download the `channels.json` file to your system from the following location.
5. Copy the file into the `<DIR>/fspm-definitions-update-tool/conf/` directory.

> **Note:** Make sure to keep the `channels.json` file name when copying. You may safely replace any previous version of the file which may already exist in the directory.

6. Make sure that the `<DIR>/fspm-definitions-update-tool/data/` directory is empty or does not exist by running the following command:

```
rm -rf <DIR>/fspm-definitions-update-tool/data/
```

7. Run the following command to create the content package:

```
<DIR>/fspm-definitions-update-tool/fspm-definitions-update-tool
```

   When the command has finished running, it prints the path for the update archive (`f-secure-updates.zip`).

8. Copy the `f-secure-updates.zip` file to a host and run the following command on the host as the `root` user to install the product updates:

```
/opt/f-secure/fsbg/bin/offline-update /PATH/TO/f-secure-updates.zip
```

   The program prepares the updates for installation and prints the channel names as the updates are extracted from the archive and then registered for installation. An example of the output for this command:

```
Preparing channel updates...
  baseguard-100-linux-x86_64.1585583363
  fsbg-100-linux-x86_64.1585581843
  linuxsecurity-1200-linux-x86_64.1585581846
  aqualnx64.1585581033
  fsbspamd-100-linux-x86_64.1582614359
  fmlibunix64.1580813614
  hydra-linux64.1585289384
Registering updates...
```

   The registered updates are installed according to the update schedule defined in the product configuration.

9. Repeat the previous step to copy and install the updates on each host where you want to install them.

> **Note:** To create another update archive with newer updates, repeat steps starting from the step 6. An empty data directory is required to create packages that are valid for use with F-Secure Linux Security.

Related Concepts

Command-line settings for automatic updates on page 39
The settings related to automatic malware definition updates that are available for `lsctl`.

Appendix A

Alert severity levels

Alerts are divided into severity levels.

Severity level	Syslog priority	Description
Informational	info	Normal operating information from the host.
Warning	warning	A warning from the host. For example, an error when trying to read a file.
Error	err	Recoverable error on the host. For example, the virus definition database update is older than the previously accepted version.
Fatal Error	emerg	Unrecoverable error on the host that requires attention from the administrator. For example, a process fails to start or loading a kernel module fails.
Security alert	alert	A security alert on the host. For example, a virus-alert. The alert includes information of the infection and the performed operation.

Appendix B

Services and logs

Topics:

- Services installed with the product
- Logs used by the product

Lists of services and logs that the product uses.

B.1 Services installed with the product

The product installs several services that are used to handle various product features.

Note: The services listed here are subject to change without advance notice.

Important: Do not manually start or stop any of the services listed here.

Active services

f-secure-baseguard-accd.service
Real-time scanner service that handles file access notifications.

f-secure-baseguard-as.service
A BaseGuard facility for email spam scanning.

Note: As of linuxsecurity channel version `12.0.286`, this service is no longer running.

f-secure-baseguard-cleanup.service
Makes sure that older channel updates are cleaned up to save disk space.

f-secure-baseguard-doormand.service
Facilitates handling subscriptions, registration and cloud service lookups.

f-secure-baseguard-icap.service
The malware analysis service used for realtime, scheduled and manual scanning.

f-secure-baseguard-orspgw.service
A local proxy for ***F-Secure's Online Reputation Service***.

f-secure-baseguard-telemetry.service
Collects information for monitoring and analysis to keep the product features working as they should.

f-secure-baseguard-tokenverify.service
An OAuth 2 authentication manager used with HTTP APIs.

This service is running, but is not used by the product.

f-secure-baseguard-update.service
Monitors F-Secure's GUTS2 service for channel updates and sends notifications to `fsbg-updated.service`. **Note**: As of BaseGuard channel version `1.0.574`, this service is no longer running.

f-secure-linuxsecurity-fsicd.service
Maintains the file integrity checker baseline.

f-secure-linuxsecurity-scand.service
Manages manual and scheduled scans.

f-secure-linuxsecurity-lspmd.service
Locally distributes centrally managed settings to product services.

f-secure-linuxsecurity-statusd.service
Collects status and statistics information from product services and relays them to the central management agent (`fsma2`).

f-secure-linuxsecurity-webserver.service
Provides an internal interface for centrally managed settings.

fsbg-pmd.service
Locally distributes centrally managed settings to BaseGuard services.

fsbg-statusd.service
Collects status and statistics information from BaseGuard services and relays them to the central management agent (`fsma2`).

fsbg-updated.service
Schedules the installation of online channel updates.

fsma2.service
Handles centrally managed settings and communication (for example, with ***Policy Manager*** or ***F-Secure Elements Endpoint Protection***).

Inactive services

These services are included during installation, but are not loaded by the product.

f-secure-baseguard-authorize.service	An OAuth 2 authorization server used with HTTP APIs.
f-secure-baseguard-sensor.service	Handles the functionality of F-Secure Elements Endpoint Detection and Response. If the Endpoint Detection and Response feature is enabled for the subscription key in use, this service is active, but it is inactive by default.
f-secure-linuxsecurity-rmmd.service	Provides status information for 3rd party RMM systems.
fsbg-enable-platform-selinux-policy.service	Turns on the SELinux confinement for product services to restrict their access to resources and actions based on the policies that are defined in the SELinux configuration.
fsbg-disable-platform-selinux-policy.service	Turns off the SELinux confinement for product services so that their access is not restricted.

Note: Both ***fsbg-{enable,disable}-platform-selinux-policy.service*** services are normally inactive, but on SELinux-capable systems, they may run temporarily after the initial product installation or after changes are made to the product configuration.

B.2 Logs used by the product

The product uses several logs for various product features.

Note: The format and contents of logs listed here are subject to change without advance notice.

systemd journal / syslog	All services log events into the syslog.
access.log	Records all log entries caused by scanning.
fsicapd.log	Records errors from the scanning service.
update.log	Records information related to updates.

Appendix C

Cloud services

The product connects to various services over the network, for example to check for updates to malware definitions.

Note: The details given in the following table are subject to change.

Service address	Protocol	Port	Accessed
`*.fsapi.com`	HTTPS	443	Occasionally, for various services
`aspam.sp.f-secure.com`	HTTPS	443	Only used in connection with antispam functionality, on service restart and very frequently during spam scanning
`guts2.sp.f-secure.com`	HTTP	80	Every hour, used for handling updates
`[*.]orsp.f-secure.com`	HTTP	80	Very frequently, triggered by reputation requests

Note: `aspam.sp.f-secure.com` is only accessed by older specific versions of the product ("linuxsecurity-2021_2" and older). The specific versions of product starting from "linuxsecurity-2021_5" no longer access this service.