Dr.WEB

Security Space (Linux)

User Manual

This document is intended for information and reference purposes regarding the Dr.Web software discussed herein. This document is not a basis for exhaustive conclusions about the presence or absence of any functional and/or technical features in Dr.Web software and cannot be used to determine whether Dr.Web software meets any requirements, technical specifications and/or parameters, and other third-party documents.

Trademarks

Disclaimer

In no event shall Doctor Web and its resellers or distributors be liable for any errors or omissions, or for any loss of profit or any other damage caused or alleged to be caused directly or indirectly by this document, or by the use of or inability to use the information contained in this document.

Dr.Web Security Space (Linux)
Version 11.1
User Manual
2/25/2025

Doctor Web Head Office

2-12A, 3rd str. Yamskogo polya, Moscow, Russia, 125124

Website: https://www.drweb.com/

Phone: +7 (495) 789-45-87

Refer to the official website for regional and international office information.

Doctor Web

Doctor Web develops and distributes Dr.Web information security solutions that provide effective protection against malicious software and spam.

Doctor Web customers include home users around the world, government agencies, small businesses, and nationwide corporations.

Since 1992, Dr.Web anti-virus solutions have been known for their continuous excellence in malware detection and compliance with international information security standards.

The state certificates and awards received by Dr.Web solutions, as well as the worldwide use of our products, are the best evidence of exceptional trust in the company products.

We thank all our customers for their support and devotion to Dr.Web products!

Table of Contents

1. Introduction

Thank you for purchasing Dr.Web Security Space. It offers reliable protection of your computer from various types of computer threats using the most advanced threat detection and neutralization technologies.

This manual is intended to help users of computers running operating systems of the GNU/Linux family (hereinafter, they will be referred to as UNIX) install and use Dr.Web Security Space.

If the previous version of Dr.Web Security Space is already installed on your computer and you wish to upgrade the product to an up-to-date version, follow the steps described in the upgrade procedure (see section Upgrading to a Newer Version).

2. Conventions and Abbreviations

The following symbols and text conventions are used in this guide:

Convention	Comment
❗	An important note or instruction.
⚠	A warning about possible errors or important notes that require special attention.
Anti-virus network	A new term or an emphasis on a term in descriptions.
<IP-address>	Placeholders.
Save	Names of buttons, windows, menu items and other program interface elements.
CTRL	Names of keyboard keys.
`/home/user`	Names of files and folders, code examples.
Appendix A	Cross-references to document chapters or internal hyperlinks to webpages.

Commands entered in the command line using a keyboard (in a terminal or a terminal emulator) are marked with the command prompt character `$` or `#` in the manual. The character indicates the privileges required for running the specified command. According to the standard convention for UNIX-based systems,

`$` indicates that the command can run with user rights;

`#` indicates that the command must run with superuser (usually *root*) privileges. To elevate the privileges, use `su` and `sudo` commands.

The list of abbreviations is provided in the Appendix H. List of Abbreviations section.

3. About This Product

In this section

Function

Dr.Web Security Space is designed to protect computers running OSes of the GNU/Linux family from viruses and other types of malware, as well as to prevent distribution of threats for various platforms.

Main components (scan engine and virus databases) are not only highly effective and resource-sparing, but also cross-platform, which lets Doctor Web experts create reliable anti-virus solutions protecting computers and mobile devices running popular operating systems from threats that target various platforms. Currently, along with Dr.Web Security Space, Doctor Web offers other anti-virus solutions for UNIX-like operating systems (GNU/Linux and FreeBSD), macOS and Windows. Moreover, other anti-virus products have been developed to deliver protection for devices running Android and Aurora OSes.

Components of Dr.Web Security Space are constantly updated, and virus databases, databases of rules for spam filtering of email messages and databases of web resource categories are regularly supplemented with new signatures to ensure up-to-date protection of computers of users, their programs and data. For additional protection against unknown malware, heuristic analysis methods implemented in the scanning engine are used. The product also uses the Dr.Web Cloud service that stores information about the latest threats, signatures of which are absent in databases.

3.1. Basic Features of Dr.Web Security Space

The basic features of Dr.Web Security Space:

1. **Detection and neutralization of threats**. Scanning for malicious programs of any kind (various viruses including those that infect mail files and boot records, trojans, email worms and so on) and unwanted software (adware, joke programs and dialers). For details on threat types, refer to Appendix A. Types of Computer Threats.

   The product uses the following methods to detect malicious and unwanted programs:

- *A signature analysis*. A scan method enabling detection of already known threats covered by virus databases.
- *A heuristic analysis*. A set of scan methods enabling detection of threats that are not known yet.
- *Cloud-based threat detection technologies* using the Dr.Web Cloud service, which collects up-to-date information about recent threats detected by various Dr.Web anti-virus products.

The heuristic analyzer may cause false-positive detections. Thus, objects that contain threats detected by the analyzer are considered “suspicious”. It is recommended that you quarantine such files and send them for analysis to the Doctor Web anti-virus laboratory. For details on the methods used to neutralize threats, refer to Appendix B. Neutralizing Computer Threats.

File system scanning can be started on demand or automatically on schedule. Both a full scan (scanning of all file system objects available to the user) and a custom scan (scanning of individual directories or files) can be performed. Furthermore, the user can start an individual scan of volume boot records and executable files that ran currently active processes. In the latter case, if a threat is detected, the malicious executable file is neutralized and all processes run from this file are forced to terminate.

For operating systems with a graphical desktop environment, integration of file scanning with either a taskbar or a graphic file manager is available. For systems that implement mandatory access control with different access levels, files that are not available for the current level can be scanned in special autonomous instance mode.

All objects containing threats detected in the file system are registered in a permanent threat registry, except those threats that were detected in autonomous instance mode.

The command-line tool supplied with Dr.Web Security Space allows scanning file systems of remote network hosts for threats. The hosts provide remote terminal access via SSH or Telnet.

Remote scanning can only be used to detect malicious and suspicious files on a remote host. To eliminate the detected threats on the remote host, use administration tools provided directly by this host. For example, for routers and other smart devices, update the firmware; for computing machines, connect to them (using a remote terminal is one of the options) and perform the necessary operations on the file system (remove or move files and so on), or run the anti-virus software installed on them.

2. **Monitoring access to files**. This mode tracks access to data files and an attempt to run executables. This allows you to detect and neutralize malware when it attempts to infect the computer. In addition to the standard monitoring mode, you can use the enhanced (or Paranoid) mode, so that the monitor blocks access to files until the scan is completed (this helps prevent access to files that contain a threat; however, a scan result becomes known only after the application accesses the file). The enhanced monitoring mode increases security, but slows down access of applications to unscanned files.

3. **Monitoring of network connections**. All attempts to access internet servers (web servers, file servers) via HTTP and FTP are monitored to block access to websites or hosts of the unwanted categories and to prevent downloading malicious files.
4. **Scanning of email messages** to prevent receiving and sending messages containing infected files and unwanted links or classified as spam.

   Scanning of email messages and downloaded files for viruses and other threats is performed on the fly. Depending on the distribution, the Dr.Web Anti-Spam component may not be included in Dr.Web Security Space. In this case, email messages are not scanned for spam.

   To detect unwanted links, Dr.Web Security Space is supplied with an automatically updated database of web resource categories and black and white lists, which are manually edited by the user. In addition, Dr.Web Security Space may also use the Dr.Web Cloud service to check whether a web resource requested by the user or a link to which is provided in an email message is classified as malicious by other Dr.Web anti-virus products.

   > If any email messages are falsely detected by the Dr.Web Anti-Spam component, we recommend you to forward them to special addresses for analysis and improvement of spam filter quality. To do that, save each message to a separate `.eml` file. Attach the saved files to an email message and forward it to the corresponding service address:
   >
   > - nonspam@drweb.com—if it contains email files *erroneously classified as spam*;
   > - spam@drweb.com—if it contains email files *erroneously not classified as spam*.

5. **Reliable isolation of infected or suspicious objects** in special storage known as quarantine to prevent any damage to the system. When quarantined, the objects are renamed according to specific rules and, if necessary, such objects can be restored to their original location only on user demand.
6. **Automatic updating** of Dr.Web virus databases and the scan engine to maintain a high level of protection against malware.
7. **Collection of statistics** on threat events, logging detected threats (available only via the command line tool) as well as sending of statistics on threat events to the Dr.Web Cloud service.
8. **Operation in centralized protection mode** to implement single security policies adopted for a network comprising this computer. It can be a corporate network, a private network (VPN) or a network of a service provider (for example, an internet service provider).

   > Since the use of the information stored by the Dr.Web Cloud service requires transferring of data about user activity (for example, addresses of visited websites), Dr.Web Cloud can be used only after the user allows it. If necessary, the use of Dr.Web Cloud can be disabled at any time in the settings of Dr.Web Security Space.

3.2. Structure of Dr.Web Security Space

Dr.Web Security Space consists of the following components:

Component	Description
Scanner	A component which performs scanning of file system objects (files, directories and boot records) for threats on user demand or on schedule. The user can start scanning either in graphical mode or in command line mode.
SpIDer Guard	A resident mode component which tracks file operations (such as creating, opening, closing and starting). It sends requests to Scanner to scan the contents of new and modified files as well as executable files when programs are started. It interacts with the OS file system using the fanotify system mechanism or a custom Linux kernel module (*LKM*, or *Loadable Kernel Module*) developed by the Doctor Web company. When using the fanotify system mechanism, the monitor can operate in enhanced or "paranoid" mode, blocking access to the files that have not been scanned yet until the scan is complete. By default, a regular monitoring mode is enabled.
SpIDer Gate	A resident mode component which monitors all network connections. • Checks whether a URL is present in databases of web resource categories or in user black lists. Blocks access to websites if URLs targeting them are included in a user black list or belong to categories marked as unwanted. • Blocks sending email messages if they contain malicious objects or unwanted links. • Sends files downloaded from the internet (from the servers access to which is not restricted) to Scanner and blocks downloading them if they contain threats. • If allowed by the user, sends requested URLs to the Dr.Web Cloud service for scanning.
Scanning Engine	A core component of the anti-virus solution. It is used by Scanner to find and detect viruses and other malicious programs as well as to analyze suspicious behavior.
Dr.Web Anti-Spam	A component which performs scanning of email messages for signs of spam. This component is not included in versions for ARM64, E2K and IBM POWER (ppc64el) architectures.
Virus databases	An automatically updated database containing information about known threats and used by the scanning engine to detect and cure them.
Database of web resource categories	An automatically updated database containing a list of web resources separated into categories and used by SpIDer Gate to block access to unwanted websites.
Updating component	A component which automatically downloads updates of virus databases and the scanning engine, databases of web resource categories from Doctor Web update servers (either on schedule or on user demand).

Component	Description
Graphical management interface	A component which provides a window graphical interface for management of Dr.Web Security Space. It allows the user to run scanning of file system objects in graphical mode, manage operation of the SpIDer Guard and SpIDer Gate monitors, view quarantined objects, start receiving updates and also configure Dr.Web Security Space operation.
Notification agent	A component which operates in background mode. It displays pop-up notifications on events and the Dr.Web Security Space indicator in the notification area, runs scheduled scanning. By default it is started together with a user session in the desktop environment.
License manager	A component which facilitates managing licenses in graphical mode. It allows to activate a license or a demo period, view information about the current license, renew it, and install or remove a license key file.

Apart from those listed in the table, Dr.Web Security Space also includes additional service components running in background with no user interaction required.

The SpIDer Guard file system monitor can operate in one of these two modes:

- *FANOTIFY*—using the fanotify system mechanism (not all GNU/Linux OSes support this mode).
- *LKM*—using the Linux loadable kernel module. The module was developed by the Doctor Web company and can be used on any GNU/Linux OS with kernel 2.6.x and later. The LKM mode is not supported for ARM64, E2K and IBM POWER (ppc64el) architectures.

By default, the file system monitor automatically chooses the appropriate operation mode according to the environment. If SpIDer Guard cannot be started, build and install the loadable kernel module from the supplied source code.

3.3. Putting in Quarantine

The quarantine of Dr.Web Security Space is a system of directories designed to isolate files containing detected threats that cannot be currently cured for some reason. For example, a detected threat can be incurable because Dr.Web Security Space is still unaware of it (for example, the threat was detected by the heuristic analyzer, but the virus databases do not cover the threat signature and a method to cure) or curing causes errors. Moreover, a file can be quarantined on user demand if the user selected the corresponding action in the list of detected threats or specified this action in settings as a reaction of Scanner or the SpIDer Guard file system monitor to threats of a specific type.

When a file is quarantined, it is renamed according to special rules to prevent its identification by users and applications and inhibit accessing it without quarantine management tools implemented in Dr.Web Security Space. Moreover, when a file is quarantined, its execution bit is always reset to prevent running this file.

Quarantine directories are located in:

- a *user home directory* (if multiple user accounts exist on the computer, a separate quarantine directory can be created for each of the users);
- a *root directory of each logical volume* mounted on the file system.

Dr.Web Security Space quarantine directories are always named `.com.drweb.quarantine` and are not created until the "**Quarantine**" (`QUARANTINE`) action is applied, that is, quarantine directories are not created until a threat is detected. At that, only a directory required for isolation of the file is created. When selecting the directory, the name of the file owner is used. Search is performed upwards from the directory containing the file to the file system root `/`; if the home directory of the owner is reached, the file is isolated in the quarantine directory under the home directory. Otherwise, the file is isolated in the quarantine directory created under the volume root directory (which is not always the same as the file system root directory). Thus, any infected file put in quarantine is always kept on the same volume, which provides for correct operation of quarantine in case there are removable data storage devices and other volumes that can be mounted in the file system occasionally and on different mount points.

A user can manage quarantine contents either in graphical mode or in command-line mode. At that, all currently available quarantine directories containing isolated objects are always processed as a single entity. From the point of view of a user who views the contents of the combined quarantine, the quarantine directory located in the home directory is a *User* quarantine, and all other quarantine directories are a *System* quarantine.

> ! You can manage quarantine even if no active license was found; however, isolated objects cannot be cured in this case.

3.4. File Permissions

To scan objects of the file system and neutralize threats, Dr.Web Security Space (or rather the user under whom it runs) requires the following permissions:

Action	Required rights
Listing all detected threats	Unrestricted. No special permission required.
Output of container contents (an archive, email file, and so on) (display only corrupted or malicious elements)	Unrestricted. No special permission required.

Action	Required rights
Moving to quarantine	Unrestricted. The user can quarantine all infected files regardless of read or write permissions on them.
Deleting threats	The user needs to have write permissions for the file that is being deleted. If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined and not deleted.
Curing	Unrestricted. The access permissions and owner of a cured file remain the same after curing. The file can be removed if deletion can cure the detected threat.
Restoring a file from quarantine	The user should have permissions to read the file and to write to the restore directory.
Deleting a file from quarantine	The user must possess write permissions to the file that was moved to quarantine.

To temporarily elevate permissions of Dr.Web Security Space running in graphical mode, you can use the corresponding button in Dr.Web Security Space window (which is available and displayed only if the privileges must be elevated to complete an operation successfully). To start the command-line management tool with superuser privileges, you can use the `su` command to change the user or the `sudo` command to run a command as another user.

3.5. Operation Modes

Dr.Web Security Space can operate both in standalone mode and as a part of a corporate or private *anti-virus network* managed by a *centralized protection server*. Such operation mode is called a *centralized protection mode*. Using this mode does not require installation of additional software or Dr.Web Security Space re-installation or uninstallation.

- In *standalone mode*, a protected computer is not connected to the anti-virus network and its operation is managed locally. In this mode, configuration and license key files are located on local disks and Dr.Web Security Space is fully managed by the protected computer. Updates of virus databases are received from Doctor Web update servers.
- In *centralized protection mode*, the protection of the computer is managed by a centralized protection server. In this mode, some functions and settings of Dr.Web Security Space can be adjusted or locked according to a general (corporate) anti-virus protection policy implemented in the anti-virus network. A custom license key file received from the selected centralized protection server to which Dr.Web Security Space is connected is used on the

computer in this mode. A license or demo key file stored on the local computer, if any, is not used. The information about Dr.Web Security Space operation, including statistics on threat events, is sent to the centralized protection server. Updates of virus databases are also received from the centralized protection server.

- In *mobile mode*, Dr.Web Security Space receives updates from Doctor Web update servers, but uses settings stored locally and a custom license key file that were received from the centralized protection server.

When Dr.Web Security Space operates in centralized protection mode or mobile mode, the following options are blocked:

- deletion of a license key file in License Manager;
- manual start of an update process and adjustment of update settings;
- configuration of file system scanning parameters.

A possibility of configuring the settings of the SpIDer Guard file system monitor as well as enabling or disabling it while Dr.Web Security Space is controlled by the centralized protection server are dependent on permissions specified on the server.

A scheduled scan is unavailable in centralized protection mode.

If starting scanning on user demand is prohibited on the centralized protection server, the page for starting scanning and the **Scanner** button on the Dr.Web Security Space window will be disabled.

Centralized Protection Concept

Doctor Web solutions for managing centralized protection use a client-server model (see the figure below).

Corporate computers or computers of clients of an IT service provider are protected by *local anti-virus components* (in this case, by Dr.Web Security Space), which ensure anti-virus protection and maintain connection to the centralized protection server.

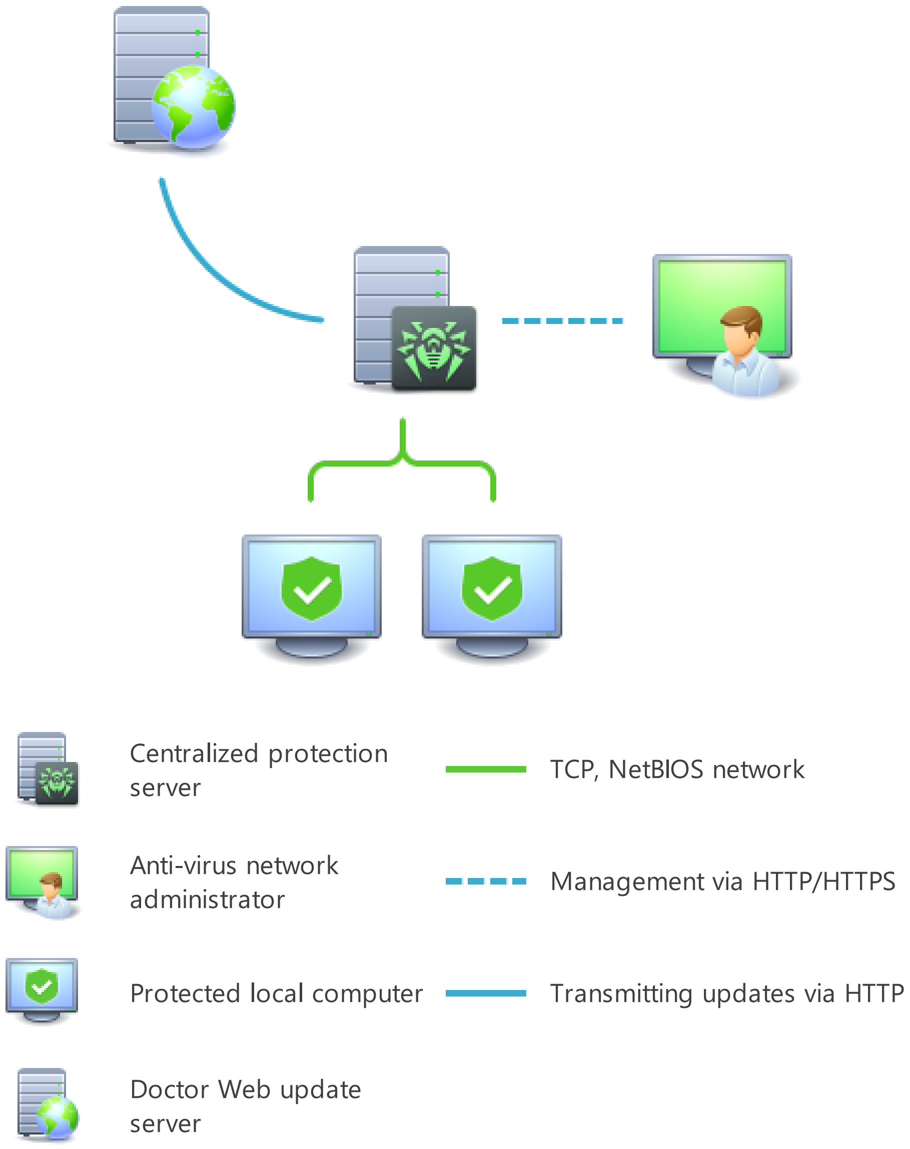

Centralized protection server
Anti-virus network administrator
Protected local computer
Doctor Web update server
TCP, NetBIOS network
Management via HTTP/HTTPS
Transmitting updates via HTTP

Figure 1. Logical structure of the anti-virus network

Local components are updated and configured from the *centralized protection server*. The entire stream of instructions, data and statistics in the anti-virus network also passes the centralized protection server. The volume of traffic between protected computers and the centralized protection server can be significant, therefore an option for traffic compression is provided. Using encryption while sending data prevents a leak of sensitive data or substitution of software downloaded onto protected computers.

All necessary updates are downloaded to the centralized protection server from Doctor Web update servers.

Changes in the configuration of local anti-virus components and command transfer are performed by anti-virus network administrators using the centralized protection server. The administrators manage configuration of the centralized protection server and topology of the anti-virus network (for example, they validate connection of a local station to the network) and configure operation of individual local anti-virus components when necessary.

> ⚠ Local anti-virus components are incompatible with anti-virus products of other companies or Dr.Web anti-virus solutions if the latter do not support operation in the centralized protection mode (for example, Dr.Web for Linux version 5.0). Installation of two anti-virus programs on the same computer can cause a system crash or a loss of important data.

The centralized protection mode allows exporting and saving Dr.Web Security Space operation reports using the centralized protection server. Reports can be exported and saved in the following formats: HTML, CSV, PDF and XML.

Connecting to the Anti-Virus Network

Dr.Web Security Space can be connected to the anti-virus network in one of the following ways:

- on the **Mode** tab of the Dr.Web Security Space configuration page;
- using the `esconnect` command of the `drweb-ctl` command-line management tool.

Disconnecting From the Anti-Virus Network

Dr.Web Security Space can be disconnected from the anti-virus network in one of the following ways:

- on the **Mode** tab of the Dr.Web Security Space configuration page;
- using the `esdisconnect` command of the `drweb-ctl` command-line management tool.

4. System Requirements and Compatibility

In this section:

- System Requirements.
- List of Supported Operating System Distributions.
- Required Additional Components and Packages.
- Compatibility with Components of Operating Systems.
- Compatibility with Security Subsystems.

System Requirements

You can use Dr.Web Security Space on a computer that meets the following requirements:

Parameter	**Requirements**
Platform	Processors of the following architectures and command systems are supported: • Intel/AMD: 32-bit (*IA-32, x86*); 64-bit (*x86-64, x64, amd64*) • ARM64 • E2K *(Elbrus)* • IBM POWER9, Power10 (*ppc64el*)
Random Access Memory (RAM)	At least 500 MB of free RAM (1 GB or more recommended).
Space on hard disk	At least 2 GB of free disk space on the volume where Dr.Web Security Space directories are stored.
Operating system	GNU/Linux based on kernel ver. 2.6.37 or later, and using PAM and `glibc` library ver. 2.13 or later, `systemd` initialization system ver. 209 or later. The supported GNU/Linux distributions are listed below.
Other	The following valid network connections: • An internet connection to download updates and for sending requests to the Dr.Web Cloud service (only if it is manually authorized by the user). • When operating in centralized protection mode, connection to the server on the local network is enough; connection to the internet is not required.

> ⚠ To enable the correct operation of the SpIDer Gate component, the OS kernel must be built with the following options:
>
> - *CONFIG_NETLINK_DIAG, CONFIG_INET_TCP_DIAG;*
> - *CONFIG_NF_CONNTRACK_IPV4, CONFIG_NF_CONNTRACK_IPV6, CONFIG_NF_CONNTRACK_EVENTS;*
> - *CONFIG_NETFILTER_NETLINK_QUEUE, CONFIG_NETFILTER_NETLINK_QUEUE_CT, CONFIG_NETFILTER_XT_MARK*.
>
> The set of required options from the specified list can depend on the GNU/Linux OS version in use.

To enable the correct operation of Dr.Web Security Space, open the following ports:

Purpose	**Direction**	**Port numbers**
To receive updates	outgoing	80
To connect to the Dr.Web Cloud service	outgoing	2075 (including those for UDP), 3010 (TCP), 3020 (TCP), 3030 (TCP), 3040 (TCP)

> ⚠ Dr.Web Security Space is incompatible with other anti-virus programs. To avoid system errors and data loss that may occur when installing two anti-viruses on one computer, uninstall all other anti-virus programs prior to the installation of Dr.Web Security Space.

List of Supported Distributions

The following GNU/Linux distributions are supported:

Platform	**Supported GNU/Linux versions**
x86_64	• ALT 8 SP • ALT Server 9, 10 • ALT Workstation 9, 10 • Astra Linux Common Edition (Orel) 2.12 • Astra Linux Special Edition 1.6 (with cumulative patch 20200722SE16), 1.7, 1.8 • CentOS 7, 8 • Debian 9, 10, 11, 12 • Fedora 37, 38 • GosLinux IC6 • Red Hat Enterprise Linux 7, 8

Platform	Supported GNU/Linux versions
	• RED OS 7.2 MUROM, RED OS 7.3 MUROM, RED OS 8 • SUSE Linux Enterprise Server 12 SP3 • Ubuntu 18.04, 20.04, 22.04, 24.04
x86	• ALT 8 SP • ALT Workstation 9, 10 • CentOS 7 • Debian 10
ARM64	• ALT 8 SP • ALT Server 9, 10 • ALT Workstation 9, 10 • Astra Linux Special Edition (Novorossiysk) 4.7 • CentOS 7, 8 • Debian 11, 12 • Ubuntu 18.04
E2K	• ALT 8 SP • ALT Server 10 • ALT Workstation 10 • Astra Linux Special Edition (Leningrad) 8.1 (with cumulative patch 8.120200429SE81) • Elbrus-D MCST 1.4 • GS CS Elbrus 8.32 TVGI.00311-28
ppc64el	• CentOS 8 • Ubuntu 20.04

⚠ Mandatory access control is not supported on Elbrus-D MCST 1.4 and GosLinux IC6.

For other GNU/Linux distributions that meet the abovementioned requirements, full compatibility with Dr.Web Security Space is not guaranteed. If a compatibility issue occurs, contact our technical support.

Required Additional Components and Packages

- To enable Dr.Web Security Space operation in graphical mode as well as to start an installer and an uninstaller in graphical mode, the X Window System graphics subsystem and any window manager are required. Moreover, the correct operation of the indicator in the Ubuntu Unity desktop environment may depend on an additional library (by default, the `libappindicator1` library is required).

- To start the installer or uninstaller designed for the command line in the graphical mode, a terminal emulator (such as xterm or xvt) is required.
- To elevate privileges during installation or uninstallation, one of the following utilities is required: `su`, `sudo`, `gksu`, `gksudo`, `kdesu`, or `kdesudo`. For correct operation of Dr.Web Security Space, PAM must be used in the operating system.

> For convenient work with Dr.Web Security Space from the command line, it is recommended to enable command auto-completion in your command shell (if disabled).
>
> If you encounter any issue with installation of additional packages and components, refer to manuals for your distribution.

Compatibility with Components of Operating Systems

- By default, the SpIDer Guard monitor uses the fanotify system mechanism, while on those operating systems on which fanotify is not implemented or is unavailable for other reasons, the component uses a custom *loadable kernel module (LKM)*, which is supplied in a pre-built form. The Dr.Web Security Space distribution has LKM modules for all GNU/Linux systems mentioned above. If required, you can build a kernel module independently from the supplied source code for any OS that uses the Linux kernel of version 2.6.x and later.

> The loadable kernel module (LKM) is not supported for ARM64, E2K and IBM POWER (ppc64el) architectures.
>
> Operation of SpIDer Guard via the LKM is not supported for operating systems started in the Xen hypervisor environment. An attempt to load the kernel module used by SpIDer Guard during the OS operation in the Xen environment can lead to a critical error of the kernel (so called “*Kernel panic*” error).
>
> SpIDer Guard can operate in enhanced or “paranoid” mode (blocks access to the files that have not been scanned yet), only via the fanotify system mechanism and providing that the OS kernel is built with the enabled `CONFIG_FANOTIFY_ACCESS_PERMISSIONS` option.

- The SpIDer Gate monitor can conflict with other firewalls installed in your system:
    - Conflict with Shorewall and SuseFirewall2 (on SUSE Linux Enterprise Server). If there is a conflict with these firewalls, an error message of SpIDer Gate with code `x109` is displayed. A way to resolve this conflict is described in the Appendix G. Known Errors section.
    - Conflict with FirewallD (on Fedora, CentOS, and Red Hat Enterprise Linux). If there is a conflict with this firewall, an error message of SpIDer Gate with code `x102` is displayed. A way to resolve this conflict is described in the Appendix G. Known Errors section.
- If the OS includes NetFilter *earlier than 1.4.15*, SpIDer Gate can operate incorrectly. The issue is related to an internal error of NetFilter: disabling SpIDer Gate causes the network to become unstable. It is recommended that you upgrade your OS to a version that includes

NetFilter 1.4.15 or later. A way to resolve this issue is described in the Appendix G. Known Errors section.

- Under normal operation, SpIDer Gate is compatible with all user applications that use network, including web browsers and mail clients. For the correct scanning of secured connections, it is necessary to add the Dr.Web Security Space certificate to a list of trusted certificates for those applications that use secure connections (for example, web browsers and mail clients).
- After adjusting the operation of SpIDer Gate (enabling the previously disabled monitor, changing the mode of scanning secure connections), it is necessary to *restart mail clients* that use IMAP to receive email messages from a mail server.

Compatibility with Security Subsystems

By default, Dr.Web Security Space does not support the SELinux security subsystem. Moreover, Dr.Web Security Space operates by default in a reduced functionality mode on the GNU/Linux systems that use mandatory access models (for example, on the systems distributed with the PARSEC mandatory access subsystem, which assigns different privilege levels, so-called mandatory levels, to users and files).

To install Dr.Web Security Space on systems with SELinux (as well as on systems that use mandatory access control models), you may have to additionally configure security subsystems to enable full functionality of Dr.Web Security Space. For details, refer to the Configuring Security Subsystems section.

5. Licensing

The rights to use Dr.Web Security Space are granted by a license purchased from the Doctor Web company or from its partners. License parameters determining user rights are set in accordance with the License agreement (see https://license.drweb.com/agreement/), which the user accepts during Dr.Web Security Space installation. The license contains information on the user and the vendor as well as usage parameters of the purchased product, including:

- a list of components licensed to the user;
- a Dr.Web Security Space license period;
- other restrictions (for example, a number of computers on which the purchased copy of Dr.Web Security Space is allowed for use).

Each Doctor Web product license has a unique serial number associated with a special file stored on the local computer. This file regulates operation of the Dr.Web Security Space components in accordance with the license parameters and is called a *license key file*.

You can activate a *demo period* for evaluation purposes. Upon activating the demo period, you gain the right to use the installed copy of Dr.Web Security Space with full functionality for this entire period. At that, a special key file named a *demo* key file is automatically generated.

If there is no valid license or a demo period is not activated (including cases when a license purchased earlier or a demo period is expired), anti-virus functions of Dr.Web Security Space are blocked. Furthermore, updates for the Dr.Web virus databases cannot be downloaded from Doctor Web update servers. However, you can activate the Dr.Web Security Space by connecting it to a centralized protection server as a part of an anti-virus network administered by an enterprise or an internet service provider. In this case, anti-virus functions and updates of the product instance installed on the computer are managed by the centralized protection server.

6. Installing and Uninstalling

This section describes how to install and uninstall Dr.Web Security Space, as well as how to obtain current updates and upgrade to a new version, if the previous version of Dr.Web Security Space is already installed on your computer.

Moreover, this section describes the procedure of custom installation and uninstallation of the components of Dr.Web Security Space (for example, to resolve errors that occurred during its operation or to install it with a limited feature set) and configuration of advanced security subsystems (such as SELinux) that could be necessary for installation or operation of Dr.Web Security Space.

- Installing Dr.Web Security Space.
- Upgrading Dr.Web Security Space.
- Uninstalling Dr.Web Security Space.
- Configuring Security Subsystems.
- Additional information:
  - Dr.Web Security Space Files Location.
  - Custom Installation and Uninstallation of Components.

To perform these procedures, superuser (the *root* user) privileges are required. To gain superuser privileges, use the `su` command to change the current user or the `sudo` command to run the specified command as a different user.

⚠ Compatibility of Dr.Web Security Space with anti-virus products of other developers is *not guaranteed*. Due to the fact that the installation of two anti-viruses on one computer can cause *errors of the operating system and a loss of important data*, it is *strongly recommended* that you uninstall anti-virus products of other developers before the installation of Dr.Web Security Space.

If your computer *already has* another Dr.Web anti-virus product installed from the universal package (`.run`), and you want to install one more Dr.Web anti-virus product (for example, you have Dr.Web Server Security Suite installed from the universal package, and in addition to it you want to install Dr.Web Security Space), make sure that the version of the installed product is the *same* as the version of Dr.Web Security Space you want to install. If the version to be installed is newer than the installed product version, *before* installation update the installed Dr.Web product to the version of the product you want to install additionally.

6.1. Installing Dr.Web Security Space

To install Dr.Web Security Space, do one of the following:

1. Download the installation file which is a universal package for UNIX systems from the Doctor Web official website. The package is supplied with installers (both graphical and console) starting depending on the environment.
2. Install Dr.Web Security Space as a set of native packages (to do that, connect to the corresponding package repository of Doctor Web).

It is recommended to install Dr.Web Security Space from the universal package on ALT 8 SP and other OSes that use outdated versions of a package manager.

After installing Dr.Web Security Space, the IMA/EVM subsystem of ALT 8 SP 11100-01 can issue a warning about potential integrity violation. To avoid this warning, run the following command after installing Dr.Web Security Space:

```
# integalert fix
```

ALT 8 SP 11100-02 and ALT 8 SP 11100-03 OSes are not subjected to this issue.

After installing Dr.Web Security Space using any one of the methods provided in this manual, you will need to activate the license or install a key file. You can also connect Dr.Web Security Space to a centralized protection server. Until that, *anti-virus protection is disabled*.

If a mail client (such as Mozilla Thunderbird) using IMAP to receive email messages is running, it is necessary to restart such mail client after installing the anti-virus to ensure scanning of incoming email messages.

After installing Dr.Web Security Space using any of the provided methods, you can uninstall or update it if fixes for its components are available or a new product version is released. If required, you can also configure GNU/Linux security subsystems for correct operation of Dr.Web Security Space. If an issue with individual components occurs, you can perform their custom installation and uninstallation without uninstalling Dr.Web Security Space.

6.1.1. Installing the Universal Package

The Dr.Web Security Space universal package is distributed as an installation file named `drweb-`*<version>*`-av-linux-`*<platform>*`.run`, where *<platform>* is the platform for which Dr.Web Security Space is intended (`x86` for 32-bit platforms, `amd64`, `arm64`, `e2s` and `ppc64el` for 64-bit platforms), for example:

```
drweb-11.1.0-av-linux-amd64.run
```

> The installation file name corresponding to the above-mentioned format is referred to as *<file_name>*`.run` below in this section.
>
> ---
>
> The version of Dr.Web Security Space is defined by the first two dot-separated figures in the name of the universal package.

To install Dr.Web Security Space components

1. Download the installation file from the official website of Doctor Web.
2. Save it to the hard disk drive of the computer to any writeable directory (for example, `/home/`*<username>*, where *<username>* is the current user name).
3. Go to the directory with the saved file and allow its execution, for example, with the following command:

   ```
   # chmod +x <file_name>.run
   ```

4. Run it with the following command:

   ```
   # ./<file_name>.run
   ```

   or use a standard file manager of your graphical shell for both changing the file properties (permissions) and running the file.

   > If you install Dr.Web Security Space on the Astra Linux SE OS of versions 1.6 and 1.7 operating in *CSE* mode, the installer can fail to start because the Doctor Web public key is not on the list of trusted keys. In this case, configure the CSE mode (see Starting in CSE Mode (Astra Linux SE 1.6 and 1.7)) and start the installer again.

   First, an integrity check of the archive is run, after which the archived files are unpacked to a temporary directory and the installer is started. If the installer is started without root privileges, it attempts to elevate its privileges asking you for the root password (the `sudo` utility is used). If the attempt fails, the installation process aborts.

> If the file system partition containing the temporary directory has not enough free space for the unpacked files, the installation process is aborted and a corresponding message is displayed. In this case, change the value of the TMPDIR system environment variable so that it points to a directory located on a partition with enough free space and restart the installation. You can also use the --target option to set the target directory (for more details, see the Custom Installation and Uninstallation of Components section).

Depending on the environment in which the distribution package is started, one of the following installers runs:

- installer for a graphical mode;
- installer for a command-line mode.

At that, the installer for the command-line mode is automatically started if the installer for the graphical mode fails to start.

5. Follow the installer instructions.

You can also start the installer in silent mode by running the command:

```
# ./<file_name>.run -- --non-interactive
```

In this case the installer is started in silent mode without displaying user interface (including dialogs of the command-line mode).

> Using this option means that you accept the terms of the Dr.Web License Agreement. After installing Dr.Web Security Space, you can find the License Agreement text in the /opt/drweb.com/share/doc/LICENSE file. The file extension indicates the language of the License Agreement. The LICENSE file (without any extension) stores the Dr.Web License Agreement in English. If you do not accept the terms of the License Agreement, you must uninstall Dr.Web Security Space after its installation.
>
> Superuser privileges are required to start the installer in silent mode. To elevate your privileges, use the su or sudo command.

> If the GNU/Linux distribution you use features the SELinux security subsystem, it can interrupt the installation process. If such situation occurs, temporarily set SELinux to the *Permissive* mode. To do this, run the following command:
>
> ```
> # setenforce 0
> ```
>
> Restart the installer afterwards. When the installation completes, configure SELinux security policies to enable correct operation of the product components.

All unpacked installation files are deleted once the installation process completes.

> It is recommended that you save the downloaded file <*file_name*>`.run`, which was used during the installation, to be able to further reinstall Dr.Web Security Space or its components if necessary without updating its version.

When the installation completes, the **Dr.Web** group appears on the **Applications** menu in your desktop graphical shell. The group comprises two items:

- **Dr.Web Security Space** to start the product in graphical mode.
- **Remove Dr.Web components** to uninstall the product.

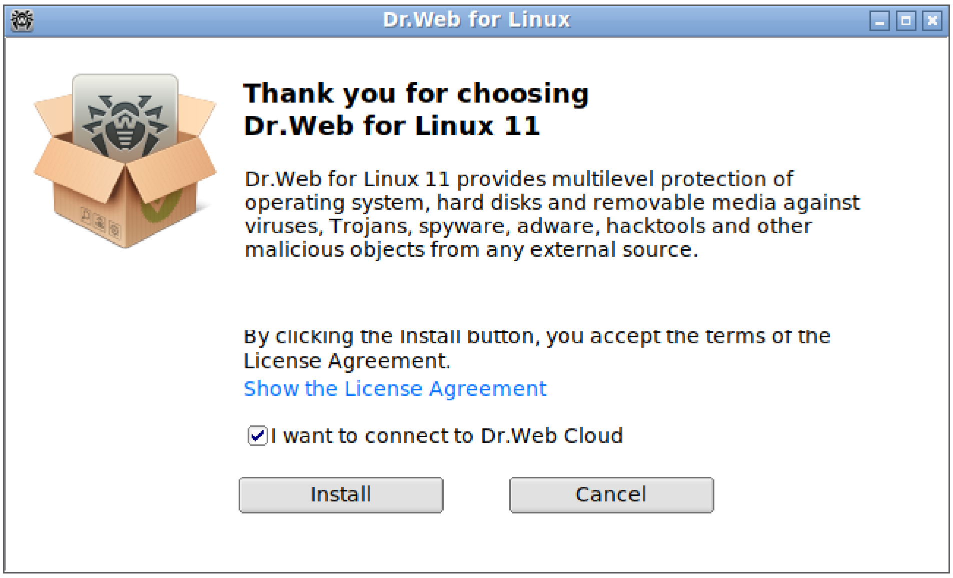
The program indicator automatically appears in the notification area after the user logs in again.

> For correct operation of Dr.Web Security Space, it may be necessary to install packages specified in the System Requirements and Compatibility section (for example, the library that enables support for 32-bit applications installed on a 64-bit platform and `libappindicator1`, which is a library for correct display of the program indicator in the notification area).

6.1.1.1. Installing in the Graphical Mode

Upon its startup, the installation program checks if there are any problems that can cause errors in Dr.Web Security Space operation or can render it inoperable. If such problems are found, an appropriate message is displayed on the screen listing the issues. You can cancel the installation by clicking **Exit** and resolve the problems. In this case, you will need to restart the installation program afterwards (after required libraries are installed, SELinux is temporarily disabled, and so on). However, you can choose not to cancel the installation of Dr.Web Security Space by clicking **Continue**. After you click the button, the process starts and the window of the installation wizard is displayed. In this case, you will need to resolve the problems after the installation completes or if errors in Dr.Web Security Space operation occur.

After the installation program for graphical mode starts, a window of the Installation Wizard displays.



Dr.Web for Linux

Thank you for choosing Dr.Web for Linux 11

Dr.Web for Linux 11 provides multilevel protection of operating system, hard disks and removable media against viruses, Trojans, spyware, adware, hacktools and other malicious objects from any external source.

By clicking the Install button, you accept the terms of the License Agreement.

Show the License Agreement

☑ I want to connect to Dr.Web Cloud

Figure 2. Welcome page of the Installation Wizard

To install Dr.Web Security Space on your computer, do the following:

1. To view the terms of the Doctor Web License agreement, click the corresponding link on the start page of the installation master. After that, a page with the License agreement text and copyright information for the installed components opens.

   When required, if a printer is installed and configured in your system, you can print off the License agreement terms and copyright information. To do that, open the corresponding tab of the License agreement page and click the **Print** button.

   To close the page, click **OK**.

2. Before the setup starts copying files, you can enable Dr.Web Security Space to connect to Dr.Web Cloud automatically after the installation. To do so, enable the corresponding option (when you start the wizard, the option is enabled by default). If you do not wish Dr.Web Security Space to use the service Dr.Web Cloud, clear the check box. If necessary, you can allow Dr.Web Security Space to connect to the Dr.Web Cloud service in the program settings at any time.

3. To continue the installation, click **Install**. By doing so, you also accept terms of Doctor Web License agreement. If you choose not to install Dr.Web Security Space on your computer, click **Cancel**. Once the button is clicked, the Installation Wizard exits.

4. After installation starts, a page with the progress bar opens. If you wish to view the logs during the installation, click **Details**.

5. After program files are successfully copied and all required adjustments to system settings are made, the final page with the installation results is displayed.

6. To exit the Installation Wizard, click **OK**. If the desktop environment you are using supports this feature, at the final installation step you will be prompted to launch Dr.Web Security Space in the graphical mode. To run the program after installation, set the **Run Dr.Web Security Space now** flag and click **OK**.

If the installation process fails due to an error, the final page of the Installation Wizard will contain the corresponding message. In this case, exit the Installation Wizard by clicking **OK**. Then remove the problems that caused this error and start an installation procedure again.

6.1.1.2. Installing from the Command Line

Once the installation program for the command line starts, the command prompt is displayed on the screen.

1. To start installation, enter *Yes* or *Y* in response to the “Do you want to continue?” question. To exit the installer, enter *No* or *N*. In this case, the installation will be canceled.
2. After that, you need to review the terms of the Doctor Web License Agreement displayed on the screen. Press ENTER to scroll down for one line or SPACEBAR to scroll down for one page.

   There are no options to scroll the License Agreement up.

3. Once you have read the License agreement, you are prompted to accept its terms. Enter *Yes* or *Y* if you accept the License agreement. If you refuse to accept it, enter *No* or *N*. In the latter case, the installer automatically exits.
4. After you accept the terms of the License Agreement, the installation of the Dr.Web Security Space components automatically starts. During the procedure, the information about the installation process (installation log), including the list of installed components, is displayed on the screen.
5. After the installation completes successfully, the installer exits automatically. If an error occurs, a message describing the error is displayed and the installer exits.
6. To start working with the installed Dr.Web Security Space, use any desirable starting method.

If the installation process fails due to an error, resolve the issues that caused this error and start the installation procedure again.

6.1.2. Installing from the Repository

Native packages of Dr.Web Security Space are stored in the Dr.Web official repository at https://repo.drweb.com. Once you have added the Dr.Web repository to the list of those used by your operating system package manager, you can install the product from native packages the same way you install any other programs from the operating system repositories. Required dependencies are automatically resolved. Futhermore, in this case an OS package manager detects updates for all Dr.Web components installed from the connected repository and suggests installing all detected updates.

To access the Dr.Web repository, internet access is required.

All the commands mentioned below, which are used to add repositories, import digital signature keys, install and uninstall packages, must be run with superuser privileges (usually, as the *root* user). To elevate your privileges, use the `su` command to change the current user or the `sudo` command to run the specified command as another user.

Steps below are for the following OSes (package managers):

- Debian, Mint, Ubuntu (apt),
- ALT (apt-rpm),
- Mageia, OpenMandriva Lx (urpmi),
- Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf),
- SUSE Linux (zypper).

Debian, Mint, Ubuntu (apt)

1. The repository for these operating systems is protected with the digital signature of Doctor Web. To access the repository, import and add the digital signature key to the package manager storage by running the command:

```
# wget https://repo.drweb.com/drweb/drweb.gpg -
O /etc/apt/trusted.gpg.d/drweb.gpg
```

The `apt-key` utility was used earlier to install the digital signature key from Doctor Web. This utility is deprecated and not recommended for usage. Furthermore, OS developers recommend using the `/etc/apt/trusted.gpg.d` directory instead of the `/etc/apt/trusted.gpg` file. For this reason, when you install Dr.Web Security Space from the repository, if the signature key is installed to the `/etc/apt/trusted.gpg` directory, the `apt-get update` command (see below) displays the following warning: `Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details`. At that, the installation of Dr.Web Security Space continues as usual and the product functionality is not affected. To avoid this warning while installing or reinstalling Doctor Web products, install the signature key as indicated above.

2. To add the repository, use the command:

```
# echo "deb https://repo.drweb.com/drweb/debian 11.1 non-free"
> /etc/apt/sources.list.d/drweb.list
```

You can complete steps 1 and 2 by downloading and installing a dedicated DEB package at https://repo.drweb.com/drweb/drweb-repo11.1.deb.

3. To install Dr.Web Security Space from the repository, use the commands:

```
# apt-get update
# apt-get install drweb-workstations
```

You can also use alternative package managers (for example, Synaptic or aptitude) to install the product. Furthermore, it is recommended to use alternative managers, such as aptitude, to solve a package conflict if it occurs.

ALT (apt-rpm)

1. To add the repository, add the following line to the file /etc/apt/sources.list.d/drweb-apt-rpm-<*arch*>.list:

   ```
   rpm http://repo.drweb.com/drweb/altlinux 11.1/<arch> drweb
   ```

   where <*arch*> represents the package architecture in use:

   - for the 32-bit version: i386;
   - for the AMD64 architecture: x86_64;
   - for the ARM64 architecture: aarch64;
   - for the E2K architecture: e2s;
   - for the IBM POWER (ppc64el) architecture: ppc64le.

2. Prior to installing Dr.Web Security Space on a computer of the E2K architecture running ALT, add the following lines to the /etc/rpmrc file:

   ```
   arch_compat: e2kv4: e2s
   arch_compat: e2k: e2s
   arch_compat: e2s: noarch
   ```

3. To install Dr.Web Security Space from the repository, run the commands:

   ```
   # apt-get update
   # apt-get install drweb-workstations
   ```

   You can also use alternative package managers (for example, Synaptic or aptitude) to install the product.

Mageia, OpenMandriva Lx (urpmi)

1. Add the repository using the command:

   ```
   # urpmi.addmedia drweb https://repo.drweb.com/drweb/linux/11.1/<arch>/
   ```

   where <*arch*> represents the package architecture in use:

   - for the 32-bit version: i386;
   - for the 64-bit version: x86_64.

2. To install Dr.Web Security Space from the repository, use the command:

```
# urpmi drweb-workstations
```

   You can also use alternative package managers (for example, rpmdrake) to install the product.

Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf)

1. Add the `drweb.repo` file with the content provided below to the `/etc/yum.repos.d` directory:

```
[drweb]
name=DrWeb - 11.1
baseurl=https://repo.drweb.com/drweb/linux/11.1/$basearch/
gpgcheck=1
enabled=1
gpgkey=https://repo.drweb.com/drweb/drweb.key
```

   If you plan to write the content indicated above to a file using such command as `echo` with redirecting of an output, the `$` symbol must be escaped: `\$`.

   You can complete step 1 by downloading and installing a dedicated RPM package at https://repo.drweb.com/drweb/drweb-repo11.1.rpm.

2. To install Dr.Web Security Space from the repository, use the command:

```
# yum install drweb-workstations
```

   On Fedora 22 and later, it is recommended to use the `dnf` manager instead of the `yum` manager, for example:

```
# dnf install drweb-workstations
```

   You can also use alternative package managers (for example, PackageKit or Yumex) to install the product.

SUSE Linux (zypper)

1. To add the repository, use the command:

```
# zypper ar https://repo.drweb.com/drweb/linux/11.1/\$basearch/ drweb
```

2. To install Dr.Web Security Space from the repository, use the commands:

```
# zypper refresh
# zypper install drweb-workstations
```

You can also use alternative package managers (for example, YaST) to install the product.

6.2. Upgrading Dr.Web Security Space

Dr.Web Security Space has two update modes:

1. Getting updates for packages and components of the current Dr.Web Security Space version (usually such updates comprise bug fixes and minor improvements in component operation).
2. Upgrading to a newer version. This updating method is used if Doctor Web has released a new version of Dr.Web Security Space with new features.

> Dr.Web Security Space allows to update virus databases and the anti-virus engine even if the protected server does not have access to the internet.

6.2.1. Getting Current Updates

After the installation of Dr.Web Security Space using any method described in the corresponding section, the package manager automatically connects to the Dr.Web package repository:

- If installation was performed from a universal package (a `.run` file) and the system uses DEB packages (for example, Debian, Mint or Ubuntu), a separate version of the `zypper` package manager is used for operation with Dr.Web packages. It is automatically installed during Dr.Web Security Space installation.

  To get and install updated Dr.Web packages with this manager, go to the `/opt/drweb.com/bin` directory and run the commands:

  ```
  # ./zypper refresh
  # ./zypper update
  ```

- In all other cases use updating commands of the package manager of your OS, for example:
  - on Red Hat Enterprise Linux or CentOS, use the `yum` command;
  - on Fedora, use the `yum` or `dnf` command;
  - on SUSE Linux, use the `zypper` command;
  - on Mageia or OpenMandriva Lx, use the `urpmi` command;
  - on ALT, Debian, Mint or Ubuntu, use the `apt-get` command.

You can also use alternative package managers developed for your operating system. If necessary, refer to the reference guide for your package manager.

If a new Dr.Web Security Space version is released, packages with its components are put into the section of the Dr.Web repository corresponding to the new product version. In this case, updating requires switching the package manager to the new Dr.Web repository section (refer to Upgrading to a Newer Version).

6.2.2. Upgrading to a Newer Version

Introductory Remarks

The upgrade procedure for Dr.Web Security Space of earlier versions to a newer version is supported. Migrate to the newer version of Dr.Web Security Space in the same way the earlier version of Dr.Web Security Space was installed:

- If the current Dr.Web Security Space version was installed from the repository, an upgrade requires updating program packages from the repository.
- If the current Dr.Web Security Space version was installed from the universal package, to upgrade the product, you need to install another universal package that contains a newer version.

> To identify how the version of Dr.Web Security Space was installed, check whether the Dr.Web Security Space executable directory contains `uninst.sh` program uninstallation script. If so, the current version of Dr.Web Security Space was installed from the universal package; otherwise it was installed from the repository.

If you cannot update Dr.Web Security Space the same way you installed it initially, uninstall your current version, and then install a new version using any convenient method. Installation and uninstallation procedures for previous versions of Dr.Web Security Space are the same as installation and uninstallation methods described in the current manual. For additional information, refer to the User manual for your current version of Dr.Web Security Space.

> Upgrading of Dr.Web Security Space from version 6.0.2 and earlier to a newer version can be performed *only* by uninstalling the outdated version of Dr.Web Security Space and installing the newer version.

If the version of Dr.Web Security Space to be updated is controlled by a centralized protection server, it is recommended that you save the address of this server. For example, to determine the address of the centralized protection server to which Dr.Web Security Space of the version later than 6.0.2 is connected, use the following command:

```
$ drweb-ctl appinfo
```

in the output provided by this command, from the line like

```
ESAgent; <PID>; RUNNING 1; Connected <address>, on-line
```

save the <*address*> part (which can look like `tcp://`<*IP address*>:<*port*>, for example: `tcp://10.20.30.40:1234`). In addition, it is recommended that you save the server certificate file.

In case of any difficulties with finding out the parameters of the current connection, refer to the Administrator Manual for the installed version of Dr.Web Security Space and to the administrator of your anti-virus network.

Upgrading Version 9.0 and Newer

Installing Universal Package as an Upgrade

Install Dr.Web Security Space from the universal package. If necessary, during the installation you will be prompted to automatically uninstall installed components of the older version of Dr.Web Security Space.

Upgrading from the Repository

To upgrade your current version of Dr.Web Security Space that was installed from the Doctor Web repository, do one of the following, depending on the required package type:

- **In case of using RPM packages (yum):**

  1. Change the repository in use (switch from the package repository of your current version to the package repository of the new version).

     > You can find the name of the repository storing the packages of the new version in the Installing from the Repository section. For details on how to switch between repositories, refer to help guides of your distribution.

  2. Install the new version of Dr.Web Security Space from the repository using the following command:

     ```
     # yum update
     ```

     or, if the dnf manager is used (such as on Fedora 22 and later):

     ```
     # dnf update
     ```

     > If an error has occurred while updating the packages, uninstall and reinstall Dr.Web Security Space. If necessary, see sections Uninstallation of Dr.Web Security Space Installed from the Repository and Installing from the Repository (paragraphs related to your OS and package manager).

- **In case of using DEB packages (apt-get)**:

1. Change the repository in use (switch from the package repository of your current version to the package repository of the new version).
2. Update the packages of Dr.Web Security Space with the following commands:

```
# apt-get update
# apt-get dist-upgrade
```

Key File Transfer

Regardless of the selected method to update Dr.Web Security Space, your license key file is installed to the appropriate location automatically to be used by the newer version of Dr.Web Security Space.

> If any issue occurs during automatic installation of the key file, you can install it manually. The license key file of Dr.Web Security Space version 9.0 and newer is stored in the `/etc/opt/drweb.com` directory. If the valid license key file is lost, contact the Doctor Web technical support.

Restoring Connection to the Centralized Protection Server

If possible, your connection to a centralized protection server will be restored automatically after the upgrade (if the version to be upgraded was connected to the centralized protection server). In case the connection has not been automatically restored, to re-establish the connection of the upgraded version of Dr.Web Security Space to the anti-virus network, use one of the following methods:

- Select the check box on the **Mode** tab of the Dr.Web Security Space settings window.
- Use the command:

```
$ drweb-ctl esconnect <address> --Certificate <path to the server certificate file>
```

> These actions require providing an address and a server public key file stored in advance.

In case of any difficulties with the connection process, contact the administrator of your anti-virus network.

Aspects of Upgrading Procedure

- If your current version of Dr.Web Security Space is active while upgrading the product from the repository, after installing the packages of the newer version of Dr.Web Security Space, the processes of the earlier version of Dr.Web Security Space remain running until the user logs off the system. At that, if Dr.Web Security Space is operating in graphical mode, the icon of the earlier version can be displayed in the notification area.

- While upgrading Dr.Web Security Space, SpIDer Gate settings can be reset to default values.
- In case of an active mail client (such as Mozilla Thunderbird) that uses the IMAP protocol to receive email messages, restart this client after the update is complete to ensure scanning of incoming email messages.

Upgrading Version 6.0.2 and Earlier

Upgrading of Dr.Web Security Space from version 6.0.2 and earlier to a newer version can be performed only by uninstalling the outdated version of Dr.Web Security Space and installing version the newer version. For additional information about uninstalling an earlier version of Dr.Web Security Space, refer to the User manual for your version of Dr.Web Security Space.

Key File Transfer

The license key file of the earlier version of Dr.Web Security Space is not installed automatically to be used by the newer version, but you can install this key file manually. The license key file of Dr.Web Security Space 6.0.2 and earlier is stored in the `/home/`*<user>*`/.drweb` directory which is marked as hidden. If the valid license key file is lost, contact the Doctor Web technical support.

> Dr.Web Security Space of the up-to-date version does not manage the quarantine of Dr.Web Security Space 9.0 and earlier. If any isolated files remain in the quarantine of the earlier version, you can retrieve or delete these files manually. Dr.Web Security Space 6.0.2 (and earlier) uses the following directories for quarantine:
>
> - `/var/drweb/infected`—system quarantine;
> - `/home/`*<user>*`/.drweb/quarantine`—user quarantine (where *<user>* is a user name).
>
> To simplify processing of quarantined files, it is recommended to revise them using the earlier version of Dr.Web Security Space before starting an upgrade.

6.3. Uninstalling Dr.Web Security Space

Depending on the method that you used to install Dr.Web Security Space, you can uninstall the product using one of two methods:

- Start the uninstaller to uninstall the universal package (in graphical or command-line mode, depending on the environment).
- Uninstall the packages installed from the Doctor Web repository using a system package manager.

6.3.1. Uninstalling the Universal Package

Dr.Web Security Space that was installed from the universal package can be uninstalled either from the application menu of the desktop environment or from the command line.

> ⚠ The uninstaller uninstalls not only Dr.Web Security Space, but also *all other* Dr.Web products installed on your computer.
>
> If any other Dr.Web products, besides Dr.Web Security Space, are installed on your computer, follow a custom component installation and uninstallation procedure to uninstall Dr.Web Security Space only instead of starting the uninstaller.

Uninstalling Dr.Web Security Space From the Application Menu

On the application menu, click the **Dr.Web** group and then **Remove Dr.Web components**. The uninstaller will be started in graphical mode.

Uninstalling Dr.Web Security Space From the Command Line

The uninstaller starts with the `remove.sh` script located in the `/opt/drweb.com/bin` directory. Thus, to uninstall Dr.Web Security Space, run the command:

```
# /opt/drweb.com/bin/remove.sh
```

After that, the uninstaller starts either in graphical mode or in command-line mode, depending on the environment.

To run the uninstaller directly from the command line, run the command:

```
# /opt/drweb.com/bin/uninst.sh
```

Uninstallation of Dr.Web Security Space is described in the corresponding sections:

- Uninstalling the Product in the Graphical Mode,
- Uninstalling from the Command Line.

You can also start the installer in silent mode without displaying the user interface (including uninstaller dialogs in command-line mode) by running the command:

```
# /opt/drweb.com/bin/remove.sh --non-interactive
```

Root privileges are required to run the uninstaller in silent mode. To elevate your privileges, use the su and sudo commands.

On ALT 8 SP you may see the following messages in the process of uninstalling the universal package:

```
/etc/init.d/drweb-configd: No such file or directory
```

These messages do not affect the system functioning. The uninstallation procedure is being performed correctly.

6.3.1.1. Uninstalling the Product in the Graphical Mode

Once the Uninstallation wizard starts in graphical mode, its welcome page is displayed.

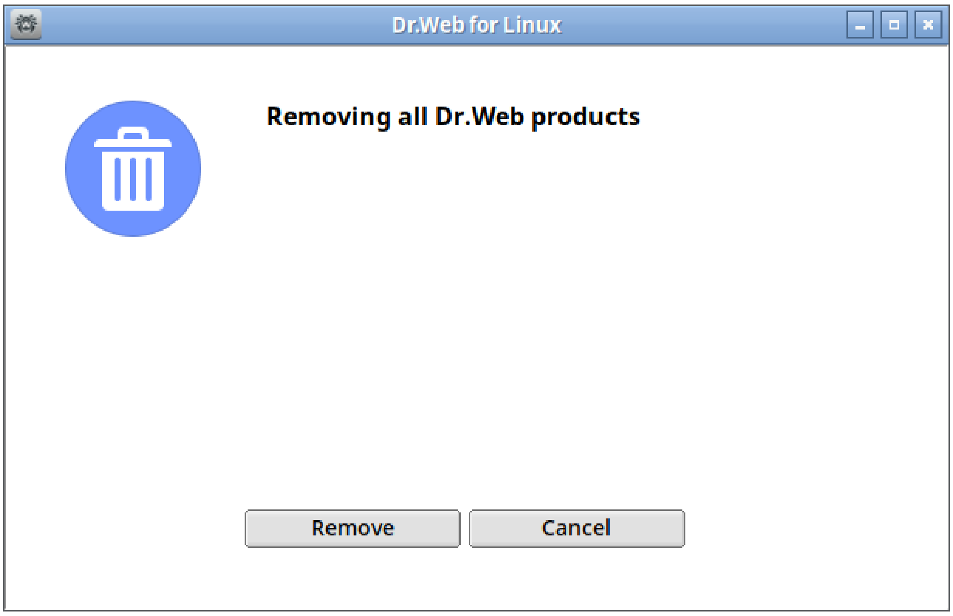


Figure 3. Welcome page

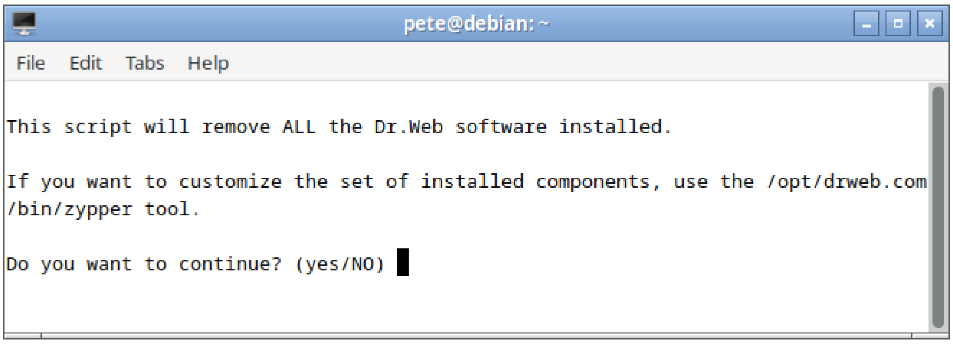
1. To uninstall Dr.Web products, click **Remove**. To close the Uninstallation Wizard and discontinue the removal of Dr.Web products from your computer, click **Cancel**.
2. After the uninstallation starts, a page with the progress bar opens. To view the log, click **Details**.

3. After Dr.Web Security Space files are successfully uninstalled and all necessary changes are made to the system settings, the Uninstallation Wizard displays the final page notifying on successful operation results.
4. To close the Uninstallation Wizard, click **OK**.

6.3.1.2. Uninstalling from the Command Line

Once the command-line uninstallation program starts, an offer to remove the product is displayed in the command line.

1. To initiate the removal, enter *Yes* or *Y* in response to the "Do you want to continue?" request. To exit the uninstaller, type *No* or *N*. In this case, removal of Dr.Web products will be canceled.



```
This script will remove ALL the Dr.Web software installed.

If you want to customize the set of installed components, use the /opt/drweb.com
/bin/zypper tool.

Do you want to continue? (yes/NO)
```

Figure 4. Offer to uninstall the product

2. An automatic uninstallation procedure of all installed Dr.Web packages will be launched after you confirm it. During this procedure, information about the removal process will be displayed on the screen and entered into the uninstallation log.
3. Once the process is completed, the uninstallation program will automatically terminate.

6.3.2. Uninstallation of Dr.Web Security Space Installed from the Repository

All commands mentioned below for package uninstallation require superuser (*root*) privileges. To elevate your privileges, use the `su` command (to change the current user) or the `sudo` command (to run a command as another user).

Steps below are for the following OSes (package managers):

- Debian, Mint, Ubuntu (apt),
- ALT (apt-rpm),
- Mageia, OpenMandriva Lx (urpmi),
- Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf),
- SUSE Linux (zypper).

Debian, Mint, Ubuntu (apt)

To uninstall the root meta-package of Dr.Web Security Space, run the command:

```
# apt-get remove drweb-workstations
```

To uninstall the root meta-package together with all dependencies, run the command:

```
# apt-get remove drweb-workstations --autoremove
```

To automatically uninstall all packages that are no longer used, you can also use the command:

```
# apt-get autoremove
```

Uninstallation using the `apt-get` command has the following aspects:

1. The first command uninstalls only the `drweb-workstations` package; all other packages that could be automatically installed to resolve dependencies remain in the system.
2. The second command uninstalls all the packages whose name starts with `drweb` (the standard name prefix for Dr.Web products). This command uninstalls all packages with this prefix, not only those of Dr.Web Security Space.
3. The third command uninstalls all packages that were automatically installed to resolve dependencies of other packages and are no longer necessary (for example, due to their uninstallation). This command uninstalls all packages that are not used, not only those of Dr.Web Security Space.

You can also use alternative package managers (for example, Synaptic or aptitude) to uninstall Dr.Web Security Space packages.

ALT (apt-rpm)

In this case, steps to uninstall Dr.Web Security Space are the same as on Debian and Ubuntu operating systems (see above).

You can also use alternative package managers (for example, Synaptic or aptitude) to uninstall Dr.Web Security Space packages.

⚠ On ALT 8 SP, the following messages can be displayed in the console during uninstallation:

```
/etc/init.d/drweb-configd: No such file or directory
```

These messages do not affect the functioning of the system. The uninstallation procedure is being performed correctly.

Mageia, OpenMandriva Lx (urpme)

To uninstall the root meta-package of Dr.Web Security Space, run the command:

```
# urpme drweb-workstations
```

To automatically uninstall all packages that are no longer used, you can use the command:

```
# urpme --auto-orphans drweb-workstations
```

⚠ Uninstallation using the `urpme` command has the following aspects:

1. The first command uninstalls only the `drweb-workstations` package; all other packages that could be automatically installed to resolve dependencies remain in the system.
2. The second command uninstalls the `drweb-workstations` root meta-package and all packages that were automatically installed to resolve dependencies of other packages and are no longer necessary (for example, due to their uninstallation). This command uninstalls all packages that are not used, not only those of Dr.Web Security Space.

You can also use alternative package managers (for example, rpmdrake) to uninstall Dr.Web Security Space packages.

Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf)

To uninstall all installed Dr.Web packages, run the following command (the * character must be escaped on some OSes: *):

```
# yum remove drweb*
```

On Fedora 22 and later, it is recommended to use the dnf manager instead of the yum manager, for example:

```
# dnf remove drweb*
```

⚠ Uninstallation using yum (dnf) has the following aspects: this command uninstalls all the packages whose name starts with drweb (the standard name prefix for Dr.Web products). This command uninstalls all packages with this prefix, not only those of Dr.Web Security Space.

You can also use alternative package managers (for example, PackageKit or Yumex) to uninstall Dr.Web Security Space packages.

SUSE Linux (zypper)

To uninstall the root meta-package of Dr.Web Security Space, run the command:

```
# zypper remove drweb-workstations
```

To uninstall all installed Dr.Web packages, run the following command (the * character must be escaped on some OSes: *):

```
# zypper remove drweb*
```

⚠ Uninstallation using the zypper command has the following aspects:

1. The first command uninstalls only the drweb-workstations package; all other packages that could be automatically installed to resolve dependencies remain in the system.
2. The second command uninstalls all the packages whose name starts with drweb (the standard name prefix for Dr.Web products). This command uninstalls all packages with this prefix, not only those of Dr.Web Security Space.

You can also use alternative package managers (for example, YaST) to uninstall Dr.Web Security Space packages.

6.4. Additional Information

6.4.1. Dr.Web Security Space Files Location

After the installation of Dr.Web Security Space, its files are located in the `/opt`, `/etc`, and `/var` directories of the file system.

Structure of the directories

Directory	**Contents**
`/opt/drweb.com`	Executable files of components and main libraries necessary for Dr.Web Security Space operation.
`/etc/opt/drweb.com`	Component setting files (by default) and a license key file for Dr.Web Security Space operation in the standalone mode.
`/var/opt/drweb.com`	Virus databases, scan engine, temporary files, and additional libraries necessary for Dr.Web Security Space operation.

6.4.2. Custom Component Installation and Uninstallation

If necessary, you can choose to install or uninstall only certain Dr.Web Security Space components by installing or uninstalling the respective packages. Perform custom installation or uninstallation the same way Dr.Web Security Space was installed.

To reinstall a component, you can uninstall it first and then install again.

Installation and uninstallation of Dr.Web Security Space components:

- installed from a repository;
- installed from a universal package.

1. Installation and Uninstallation of Dr.Web Security Space Components Installed from a Repository

If Dr.Web Security Space is installed from a repository, to install or uninstall a component, use a respective command of the package manager of your OS, for example:

1. To remove the Dr.Web Updater component (the `drweb-update` package) from Dr.Web Security Space installed on CentOS, use the command:

```
# yum remove drweb-update
```

2. To add the Dr.Web Updater component (the `drweb-update` package) to Dr.Web Security Space installed on Ubuntu, use the command:

```
# apt-get install drweb-update
```

If necessary, use help on the package manager of your OS.

2. Installation and Uninstallation of Dr.Web Security Space Components Installed from a Universal Package

If Dr.Web Security Space is installed from a universal package and you want to additionally install or reinstall a package of a component, you will need an installation file (with the `.run` extension) that was used to install Dr.Web Security Space. If you do not have this file anymore, download it from the Doctor Web company website.

Unpacking the Installation File

When you start the `.run` file, you can specify the following command-line parameters:

`--noexec`—unpack Dr.Web Security Space installation files instead of starting the installation process. The files will be placed in the directory that is specified in the `TMPDIR` environment variable (usually, `/tmp`).

`--keep`—do not delete Dr.Web Security Space installation files and the installation log after the installation.

`--target` *<directory>*—unpack Dr.Web Security Space installation files to the specified *<directory>*.

To get a full list of command-line parameters that can be used with the installation file, run the command:

```
$ ./<file_name>.run --help
```

For custom installation of Dr.Web Security Space components, go to the directory containing the unpacked Dr.Web Security Space package files. If there is no such directory, run the command:

```
$ ./<file_name>.run --noexec --target <directory>
```

As the result, a nested *<file_name>* directory containing the unpacked Dr.Web Security Space files will appear in the *<directory>* directory.

Custom Installation of Components

The `.run` installation file contains packages of all Dr.Web Security Space components (in the RPM format) and auxiliary files. Package files of each component are named as follows:

```
<component_name>_<version>~linux_<platform>.rpm
```

where *<version>* is a string that contains the version and date of a package release, and *<platform>* is a platform to run Dr.Web Security Space. Names of all Dr.Web Security Space component packages start with the `drweb` prefix.

The installation kit includes the `zypper` package manager intended for the installation of packages. For custom installation, use the `installpkg.sh` script. To do that, firstly unpack the contents of the installation package to any writeable directory.

> To install packages, superuser (the *root* user) privileges are required. To gain superuser privileges, use the `su` command to change the current user or the `sudo` command to run the specified command as a different user.

To install a component package, go to the directory containing the unpacked installation kit and run the following command from the console or from a terminal emulator:

```
# ./scripts/installpkg.sh <package_name>
```

For example:

```
# ./scripts/installpkg.sh drweb-update
```

If it is necessary to install Dr.Web Security Space in full, start the installation script by running the command:

```
$ ./install.sh
```

Furthermore, you can install all Dr.Web Security Space packages (among other things, to install missing or accidentally removed components) by starting the installation of the root meta-package:

```
# ./scripts/installpkg.sh drweb-workstations
```

Custom Uninstallation of Components

If your OS uses RPM packages, to uninstall a package of a component, use the corresponding uninstallation command of the package manager of your OS:

- on Red Hat Enterprise Linux and CentOS, use the `yum remove` *<package_name>* command;

- on Fedora, use the `yum remove` *<package_name>* or `dnf remove` *<package_name>* command;
- on SUSE Linux, use the `zypper remove` *<package_name>* command;
- on Mageia or OpenMandriva Lx, use the `urpme` *<package_name>* command;
- on ALT, use the `apt-get remove` *<package_name>* command.

For example, on Red Hat Enterprise Linux:

```
# yum remove drweb-update
```

If your OS uses DEB packages, use the `zypper` package manager supplied with Dr.Web Security Space, for the custom uninstallation. To do that, go to the `/opt/drweb.com/bin` directory and run the command:

```
# ./zypper rm <package_name>
```

For example:

```
# ./zypper rm drweb-update
```

If you want to uninstall Dr.Web Security Space, start the uninstallation script by running the command:

```
# ./uninst.sh
```

To reinstall a component, you can uninstall it first and then install again having started custom or full installation from the installation kit.

6.5. Configuring Security Subsystems

Presence of the SELinux enhanced security subsystem in the OS as well as the use of mandatory access control systems, such as PARSEC—as opposed to the classical discretionary model used by UNIX—causes issues in the operation of Dr.Web Security Space when its default settings are used. To ensure correct operation of Dr.Web Security Space in this case, it is necessary to make additional changes to the settings of the security subsystem and/or to the settings of Dr.Web Security Space.

This section covers the following settings that enable the correct operation of Dr.Web Security Space:

- Configuring SELinux Security Policies.
- Configuring the permissions of the PARSEC mandatory access control system (the Astra Linux SE OS).
- For ALT 8 SP and other distributions using pam_namespace.
- Configuring the launch in the CSE (Closed Software Environment) mode (Astra Linux SE 1.6 and 1.7).

> Configuring the permissions of the PARSEC mandatory access control system for Dr.Web Security Space allows the anti-virus components to bypass the restrictions of set security policies and get access to files of different privilege levels.

Note that even if you have not configured the permissions of the PARSEC mandatory access control system for Dr.Web Security Space components, you still will be able to start file scanning via the Graphical management interface of Dr.Web Security Space in the autonomous instance mode. For that, run the `drweb-gui` command with the `--Autonomous` parameter. You can also launch the scanning directly from the command line. To do this, run the `drweb-ctl` command with the same parameter (`--Autonomous`). In this case, it will be possible to scan files that require a level of privileges not higher than the level of the user who started the scanning session. This mode has the following aspects:

- To run it as an autonomous instance, you will need a valid key file, operating in a centralized protection mode is not supported (it is possible to install the key file exported from a centralized protection server). In this case, even if Dr.Web Security Space is connected to the centralized protection server, the autonomous instance *does not notify* the centralized protection server of the threats detected in the autonomous instance mode.
- All supplementary components that maintain the operation of the autonomous instance will be started as the current user and will operate with a custom configuration file.
- All temporary files and UNIX sockets used for interaction of components are created only in a directory with an unique name. This directory is created by the started autonomous instance in the directory for temporary files (specified by the `TMPDIR` environment variable).
- The autonomous instance of the graphical management interface *does not start* the SpIDer Guard and SpIDer Gate monitors, only file scanning and quarantine management functions supported by Scanner are available.

- All the required paths (to virus databases, scanning engine and executable files of service components) are set to default values or retrieved from custom environment variables.
- The number of the autonomous instances working simultaneously is not limited.
- When the autonomous instance is shut down, the set of components maintaining it is also terminated.

6.5.1. Configuring SELinux Security Policies

If your GNU/Linux distribution features the SELinux (*Security-Enhanced Linux*) security subsystem, you may need to adjust SELinux security policies to enable correct operation of Dr.Web Security Space service components (for example, of the scanning engine) after their installation.

1. Universal Package Installation Issues

If SELinux is enabled, the installation of the Dr.Web Security Space universal package from the installation file (`.run`) can fail because an attempt to create the *drweb* special user, as which Dr.Web Security Space components run, will be blocked.

If installation of Dr.Web Security Space from the installation file (`.run`) fails due to inability to create the *drweb* user, check the SELinux operation mode with the `getenforce` command. The command outputs the current protection mode:

- *Permissive*—protection is active but a permissive strategy is used: actions that violate the security policy are only registered in an audit log but not blocked;
- *Enforced*—protection is active and a restrictive strategy is used: actions that violate security policies are blocked and registered in the audit log;
- *Disabled*—SELinux is installed but not active.

If SELinux is operating in the *Enforced* mode, temporarily (during the installation of Dr.Web Security Space) change its mode to *Permissive*. For that purpose, use the following command:

```
# setenforce 0
```

which temporarily (until the next restart) enables the *Permissive* mode of SELinux.

Regardless of the operation mode enabled with the `setenforce` command, after the restart of the operating system, SELinux returns to the safe operation mode specified in its settings (the file with SELinux settings is usually stored in the `/etc/selinux` directory).

After Dr.Web Security Space is successfully installed from the installation file, enable the *Enforced* mode again before starting and activating the product. For that purpose, use the following command:

```
# setenforce 1
```

2. Dr.Web Security Space Operation Issues

In some cases when SELinux is enabled, certain auxiliary Dr.Web Security Space components (for example, drweb-se and drweb-filecheck used by Scanner and SpIDer Guard) cannot start. If so, object scanning and file system monitoring become unavailable. When these auxiliary modules fail to start, messages about *119* and *120* errors are displayed on the main Dr.Web Security Space window and logged by syslog (the log is usually stored in the /var/log/ directory).

When the SELinux security system blocks access, such an event is also output to an audit system log. In general, when the audit daemon is used in the system, the audit log is stored in the /var/log/audit/audit.log file. Otherwise, messages about blocked operations are written to the general log file (/var/log/messages or /var/log/syslog).

If auxiliary modules do not function because they are blocked by SELinux, compile custom security policies for them.

Certain GNU/Linux distributions do not feature the utilities mentioned below. If so, you may need to install additional packages containing them.

Creating SELinux Security Policies:

1. Create a new file with the SELinux policy source code (a .te file). This file defines restrictions related to the described module. The policy source code file can be created in one of the following ways:

   1) Using the audit2allow utility, which is the simplest method. The utility generates permissive rules from messages on access denial in system log files. You can set to search messages automatically or specify a path to the log file manually.

You can use this method only if Dr.Web Security Space components have violated SELinux security policies and these events have been registered in the audit system log. If not, wait for such an incident triggered by Dr.Web Security Space to occur or force-create permissive policies by using the policygentool utility (see below).

The audit2allow utility is contained either in the policycoreutils-python package or in the policycoreutils-devel package (for Red Hat Enterprise Linux, CentOS, Fedora operating systems, depending on the version) or in the python-sepolgen package (for Debian and Ubuntu operating systems).

Example of using audit2allow:

```
# grep drweb-se.real /var/log/audit/audit.log | audit2allow -M drweb-se
```

In this example, the audit2allow utility searches the audit.log file for access denial messages for the drweb-se module.

The utility creates two files: the drweb-se.te policy source file and the drweb-se.pp policy module ready to install.

If no corresponding incidents are found in the system audit log, the utility returns an error message.

In most cases, you do not need to modify the policy file created by the audit2allow utility; thus, it is recommended to go to step 4 for the installation of the drweb-se.pp policy module.

> ! The audit2allow utility outputs the semodule command with all arguments. By copying it to the command line and running it, you complete step 4. Go to step 2 only if you want to modify the security policies that were automatically generated for Dr.Web Security Space components.

2) Using the policygentool utility. For that purpose, specify a name of the module, the operation with which you want to configure, and the full path to its executable file.

> ! The policygentool utility included in the selinux-policy package for Red Hat Enterprise Linux and CentOS may not function correctly. If so, use the audit2allow utility.

Example of policy creation using policygentool:

- For drweb-se:

```
# policygentool drweb-se /opt/drweb.com/bin/drweb-se.real
```

- For drweb-filecheck:

```
# policygentool drweb-filecheck /opt/drweb.com/bin/drweb-filecheck.real
```

You will be prompted to specify several common domain characteristics. After that, three files that determine the policy are created for each of the modules: *<module_name>.te*, *<module_name>.fc* and *<module_name>.if*.

2. If required, edit the generated policy source file *<module_name>.te*, then use the checkmodule utility to create a binary representation (a .mod file) of this source file of the local policy.

> ! This command requires the checkpolicy package to be installed in the system.

Usage example:

```
# checkmodule -M -m -o drweb-se.mod drweb-se.te
```

3. Create a policy module for installation (a .pp file) with the help of the semodule_package utility.

Example:

```
# semodule_package -o drweb-se.pp -m drweb-se.mod
```

4. To install the created policy module, use the `semodule` utility.

   Example:

```
# semodule -i drweb-se.pp
```

For details on SELinux operating principles and configuration, refer to documentation for your GNU/Linux distribution.

6.5.2. Configuring the PARSEC Permissions

In GNU/Linux distributions having the PARSEC security subsystem, the access of all applications to files depends on their privilege level. Thus, SpIDer Guard can intercept file access events by default to the extent allowed by its privilege level.

Moreover, if the user works at any privilege level other than the zero, the graphical interface of Dr.Web Security Space cannot interact with SpIDer Guard and the anti-virus service components if they operate at different privilege levels; moreover, access to the consolidated quarantine may become unavailable.

If your OS uses PARSEC and there are accounts of users working at privilege levels other than zeroth, you need to customize Dr.Web Security Space to ensure that its components can interact while running at different privilege levels.

This section covers the following PARSEC settings that enable correct operation of Dr.Web Security Space:

- Customizing the interaction of components running at different privilege levels.
- Customizing the automatic start of Dr.Web Security Space components with user privileges.
- Configuring SpIDer Guard to intercept file access events.

> To perform these procedures, superuser (the *root* user) privileges are required. To gain superuser privileges, use the `su` command to change the current user or the `sudo` command to run the specified command as another user.

Customizing the interaction of components running at different privilege levels

The *privsock* mechanism is designed to enable the operation of system network services that do not process information using the mandatory context but interact with processes that operate in the mandatory context of an access subject. `drweb-configd` is the Dr.Web Security Space service component that is responsible for interaction of all anti-virus components among each other.

To grant the configuration management daemon of Dr.Web Security Space (`drweb-configd`) a privilege to use *privsock*, it is necessary to edit the `/etc/parsec/privsock.conf` system file. To change settings, we recommend that you use the `drweb-configure` configuration tool bundled with Dr.Web Security Space, or you can make manual changes to the required configuration files.

1. Using the drweb-configure tool

The required changes will be made automatically after running the following command:

```
# drweb-configure session <mode>
```

where <*mode*> may have one of the following values:

- `enable`—use *privsock*;
- `disable`—do not use *privsock*.

2. Editing configuration files manually

For Astra Linux SE version 1.6 and later

1. Open the `/etc/parsec/privsock.conf` file in any text editor. Add the following lines to this file:

```
/opt/drweb.com/bin/drweb-configd
/opt/drweb.com/bin/drweb-configd.real
```

2. Save the file and restart the operating system.

Customizing the automatic start of the components with user privileges

To make Dr.Web Security Space components with which the user interacts available in the user environment (when the user works at a privilege level other than zero), you need to make changes to the files containing PAM settings to ensure the automatic start of the required Dr.Web Security Space components at the beginning of the user session and their termination at the end of the session. The module (the custom `pam_drweb_session.so` PAM module by Doctor Web starts the `drweb-session` mediation component, which connects the local copies of components running in the user environment to the components operating with zero-level privileges and running automatically at the OS startup).

To change PAM settings, we recommend that you use the `drweb-configure` configuration tool included in Dr.Web Security Space; alternatively, you can edit the required configuration files manually.

1. Using the drweb-configure tool

To make configuring complex parameters of Dr.Web Security Space more convenient, we have developed a dedicated auxiliary tool `drweb-configure`.

1. To enable or disable the automated start of the required Dr.Web Security Space components in the environment of the user who has a privilege level other than zero, use the following command:

   ```
   # drweb-configure session <mode>
   ```

   where <*mode*> may have one of the following values:

   - `enable`—enable the automated start of the necessary components during the user session with user privileges.
   - `disable`—disable the automated start of the required components during the user session with user privileges (this will render a number of Dr.Web Security Space functions unavailable).

2. Restart the operating system.

To use help on how to use `drweb-configure` for configuring PAM settings, run the following command:

```
$ drweb-configure --help session
```

2. Editing PAM configuration files manually

For Astra Linux and other distributions using the pam_parsec_mac.so PAM module

1. To change PAM configuration, you need to edit all configuration files in the `/etc/pam.d` directory that call the `pam_parsec_mac.so` PAM module. You can get the full list of such files by performing the following command:

   ```
   # grep -R pam_parsec_mac.so /etc/pam.d
   ```

   Add the following records of the *session* type to all files from the list:

   - Above the first record of the *session* type:

   ```
   session optional pam_drweb_session.so type=close
   ```

   - After the last record of the *session* type:

   ```
   session optional pam_drweb_session.so type=open
   ```

2. Save the changed files.

3. Create a symbolic link to the pam_drweb_session.so file from the system directory containing PAM modules. The pam_drweb_session.so file is located in the Dr.Web Security Space library directory /opt/drweb.com/lib/ (for 64-bit operating systems, for instance, the path to the module is /opt/drweb.com/lib/x86_64-linux-gnu/pam/).
4. Restart the operating system.

Configuring SpIDer Guard to intercept file access events

To give the SpIDer Guard file monitor an ability to detect the attempts of accessing files, which have any level of access privileges, you need to switch SpIDer Guard to the *Fanotify* operating mode.

To switch SpIDer Guard to the *Fanotify* operating mode, run the following command:

```
# drweb-ctl cfset LinuxSpider.Mode Fanotify
```

To get additional information, run the following command:

```
$ man drweb-spider
```

6.5.3. For ALT 8 SP and other distributions using pam_namespace

To make Dr.Web Security Space components with which the user interacts available in the user environment, you need to make changes to the files containing PAM settings to ensure the automatic start of the required Dr.Web Security Space components at the beginning of the user session and their termination at the end of the session.

⚠ On ALT 8 SP, changes are required for all privilege levels, and on other OSes using pam_namespace—when the user works at a privilege level other than zero.

The custom pam_drweb_session.so PAM module developed by the Doctor Web company starts the drweb-session mediation component, which connects the local instances of components running in the user environment to the components operating with zero-level privileges and running automatically at OS startup.

To change PAM settings, we recommend that you use the drweb-configure configuration utility included in Dr.Web Security Space; alternatively, you can edit the required configuration files manually.

Perform the following actions before introducing changes on ALT 8 SP 11100-02:

1. Log in as *officer*.
2. Gain superuser rights:

   ```
   $ su -
   ```

3. Install the policy:

   ```
   # semodule -i /opt/drweb.com/share/drweb.pp
   ```

4. Update file security contexts on the basis of the installed policy:

   ```
   # restorecon -r /opt/drweb.com
   ```

1. Using the drweb-configure tool

To make configuring complex parameters of Dr.Web Security Space more convenient, we have developed a dedicated auxiliary utility `drweb-configure`.

1. To enable or disable the automated launch of the required Dr.Web Security Space components in the environment of the user who has a privilege level other than zero, use the following command:

   ```
   # drweb-configure session <mode>
   ```

   where <*mode*> may have one of the following values:

   - `enable`—enable the automated launch of the necessary components during the user session with user privileges.
   - `disable`—disable the automated launch of the required components during the user session with user privileges (this will render a number of Dr.Web Security Space functions unavailable).

2. Restart the operating system.

To use help on how to use `drweb-configure` for configuring PAM settings, run the following command:

```
$ drweb-configure --help session
```

2. Editing PAM configuration files manually

1. To change PAM configuration, you need to edit all configuration files in the `/etc/pam.d` directory that call the `pam_namespace.so` PAM module. You can get the full list of such files by performing the following command:

```
# grep -R pam_namespace.so /etc/pam.d
```

Add the following records of the *session* type to all files from the list:

- Above the first record of the *session* type:

```
session optional pam_drweb_session.so type=close
```

- After the last record of the *session* type:

```
session optional pam_drweb_session.so type=open
```

2. Save the changed files.
3. Create a symbolic link to the `pam_drweb_session.so` file from the system directory containing PAM modules. The `pam_drweb_session.so` file is located in the Dr.Web Security Space library directory `/opt/drweb.com/lib/` (for 64-bit operating systems, for instance, the path to the module is `/opt/drweb.com/lib/x86_64-linux-gnu/pam/`). A command example for 64-bit ALT 8 SP OS:

```
# ln -s /opt/drweb.com/lib/x86_64-linux-
gnu/pam/pam_drweb_session.so /lib64/security/pam_drweb_session.so
```

Perform the following additional actions on ALT 8 SP 11100-02 and ALT 8 SP 11100-03:

1. In the `/etc/pam.d/newrole` file, replace

```
session optional pam_drweb_session.so type=close
```

with the following:

```
session optional pam_drweb_session.so type=cleanup
```

2. Edit the `/etc/pam.d/su` and `/etc/pam.d/sudo` files by adding the following string to the end:

```
session optional pam_drweb_session.so type=close
```

3. Save the changed files.
4. Run the command:

```
# cp /opt/drweb.com/share/drweb-session/drweb-
session.sh /etc/profile.d/
```

4. Restart the operating system.

6.5.4. Starting in CSE Mode (Astra Linux SE 1.6 and 1.7)

The Astra Linux SE OS supports a special *closed software environment* (CSE) mode. In this mode, applications can be started only if their executable files are signed with a digital signature of a developer whose public key is added to the OS list of trusted keys.

Starting with Astra Linux SE 1.6, the signing mechanism has been changed. You must configure Astra Linux SE 1.6 and 1.7 prior to starting Dr.Web Security Space in CSE mode.

To configure Astra Linux SE 1.6 and 1.7 to start Dr.Web Security Space in CSE mode

1. Install the `astra-digsig-oldkeys` package, if not installed, using an OS installation disk.
2. Add the public key of the Doctor Web company to the directory `/etc/digsig/keys/legacy/keys` (if the directory is absent, create it):

   ```
   # cp /opt/drweb.com/share/doc/digsig.gost.gpg /etc/digsig/keys/legacy/keys
   ```

3. Run the command:

   ```
   # update-initramfs -k all -u
   ```

4. Restart the operating system.

7. Getting Started

1. Activate Dr.Web Security Space.
2. Ensure its proper operation.
3. Set the file monitoring mode.
4. Specify exclusions, if any.

7.1. Registration and Activation

In this section:

- Purchasing and Registering Licenses.
- Dr.Web Security Space Activation:
  - Demo Period.
  - Key File Installation.
  - Connection to the Centralized Protection Server.
- Repeated Registration.

Purchasing and Registering Licenses

After a license is purchased, updates to product components and virus databases can be regularly downloaded from Doctor Web update servers. Moreover, standard technical support provided by Doctor Web and its partners becomes available.

You can purchase any Dr.Web product as well as obtain a product serial number either from our partners (see the list of partners on https://partners.drweb.com/) or in our online store https://estore.drweb.com/. For details on license options, visit the Doctor Web official website at https://license.drweb.com/.

License registration is required to prove that you are a legal user of Dr.Web Security Space and activate its anti-virus functions, including regular updates of virus databases. We recommend that you register the product and activate the license once the installation is completed.

Dr.Web Security Space Activation

A license can be activated one of the following ways:

- Via the Registration Wizard included in License Manager.
- On the Doctor Web official website at https://products.drweb.com/register/.

To activate or renew the license, you need to enter the serial number. The serial number is supplied with Dr.Web Security Space or via email when purchasing or renewing the license online.

If you have several licenses for using Dr.Web Security Space on several computers, but choose to use Dr.Web Security Space only on one computer, you can specify this and, hence, all licenses will be combined and the license validity period will be automatically extended.

Demo Period

Users of Dr.Web products can obtain a demo period for 1 month. It can be received in the Registration wizard window of License Manager without providing personal data.

The Registration Wizard of License Manager opens upon the first Dr.Web Security Space startup (usually the Registration Wizard starts once the installation is complete). You can start registration from the License Manager window at any time by clicking **Get new license** on the page with information on the current license.

To activate a license using the serial number or request a demo license, a valid internet connection is required.

When a demo period or license is activated via License Manager, the key file (license or demo) is automatically generated on the local computer in its target directory. If you register on the website, the key file will be sent to you by email and you will need to install it manually.

If the registration wizard is unavailable (for example, the operating system has no GUI), you can use the `license` command of the `drweb-ctl` command-line utility, which allows you to obtain the license key file corresponding to the serial number of the registered license.

Key File Installation

If you have a key file corresponding to a valid license for the product (for example, you have obtained the key file from a vendor by email after registration or Dr.Web Security Space is being migrated to another computer), you can activate Dr.Web Security Space by specifying a path to the key file. You can do that as follows:

- In the License Manager by clicking **Other activation types** on the first step of the registration procedure and specifying a path to the key file or to the `.zip` archive with the key.
- Manually. For that:
  1. Unpack the key file if archived.
  2. Copy the key file to the `/etc/opt/drweb.com` directory and rename the file to `drweb32.key`.
  3. Run the command:

     ```
     # drweb-ctl reload
     ```

     to apply changes.

You can also use the following command:

```
# drweb-ctl cfset Root.KeyPath <path to the key file>
```

> ⚠ In the latter case, the key file will not be copied to the `/etc/opt/drweb.com` directory and will remain in its original location, moreover, the file name can be different from `drweb32.key`.
>
> If the key file is not copied to the `/etc/opt/drweb.com` directory, the user becomes responsible for ensuring that the file is protected from corruption or deletion. This installation method is not recommended as the key file can be accidentally deleted (for example, if the directory, where the key file is located, is automatically cleaned up by the system). Remember that if a key file is lost, you can request a new one, but the number of such requests is limited.

Connection to the Centralized Protection Server

If the internet service provider or network administrator submits a file with settings for connecting to the centralized protection server, you can activate Dr.Web Security Space by specifying the file path. This can be done as follows:

- In the open program settings window, go to the **Mode** tab and select the **Enable centralized protection mode** check box. On the drop-down list, select the **Load from file** item, specify the path to the connection settings file and click **Connect**.

Repeated Registration

If a key file is lost but the license is not expired, you must register again by providing the personal data you specified during the first registration. You can use a different email address. In this case, the license key file will be sent at the new address.

A license key file can be obtained using the License Manager or the license management command a limited number of times. If that amount has been exceeded, you can confirm the registration of your serial number at https://products.drweb.com/register/ to receive the key file. The key file is sent to the email that was specified during the first registration.

7.1.1. Key File

The key file is a special file stored on the local computer. It corresponds to the purchased license or activated demo period for Dr.Web Security Space. The file contains information on the provided license or demo period and regulates usage rights in accordance with it.

The key file has `.key` extension and is valid if satisfies the following criteria:

- License or demo period is not expired.
- Demo period or license applies to all anti-virus components required by the product.
- Integrity of the key file is not violated.

If any of the conditions are violated, the license key file becomes invalid.

> During Dr.Web Security Space operation, the key file must be located in the default directory `/etc/opt/drweb.com` under the name `drweb32.key`.
>
> Components of Dr.Web Security Space regularly check whether the key file is available and valid. The key file is digitally signed to prevent its editing. So, the edited key file becomes invalid. It is not recommended to open your key file in text editors in order to avoid its accidental invalidation.
>
> If no valid key file (license or demo) is found, or if the license is expired, operation of the anti-virus components is blocked until a valid key file is installed.

It is recommended to keep the license key file until it expires, and use it to reinstall Dr.Web Security Space or install it on a different computer. In this case, you must use the same product serial number and customer data that you provided during the registration.

> Dr.Web key files are usually packed in a ZIP archive if sent via email. The archive with a key file is named `agent.zip` (note that if there is *several* archives in an email message, you should use only `agent.zip`). In the Registration Wizard, you may specify the direct path to the archive without its unpacking. Before installing a key file, unpack it using any suitable tool and extract a key file to any directory (for example, to your home directory or to a USB flash drive).

7.1.2. Connection Settings File

The connection settings file is a special file that stores parameters that configure connection between Dr.Web Security Space and the centralized protection server. This file is supplied by the administrator of the anti-virus network or the internet service provider (if the latter provides support for the central anti-virus protection service).

You can use this file to activate Dr.Web Security Space when connecting it to the centralized protection server (in this case, you cannot use Dr.Web Security Space in the standalone mode without purchasing additional license).

7.2. Testing Product Operation

The *EICAR (European Institute for Computer Anti-Virus Research)* test helps testing operation of anti-virus programs that detect viruses using signatures. This test was designed specifically so that users could test reaction of an installed anti-virus to a threat without putting their computers at risk.

Although the *EICAR* test program is not actually malware, it is treated by the majority of anti-viruses as a virus. Dr.Web anti-virus products report the following upon detection of this “virus”: *EICAR Test File (NOT a Virus!)*. Other anti-viruses alert users in a similar way. The EICAR test program is a 68-byte `.com` file for MS-DOS/Windows that outputs the following message to the console or to a terminal emulator screen when running:

```
EICAR-STANDARD-ANTIVIRUS-TEST-FILE!
```

The test program body contains only text characters that form the following string:

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

If you create a text file consisting of the string provided above, the resulting file will be the “virus” program.

If Dr.Web Security Space operates correctly, this file must be detected during a file system scan regardless of the scan type and the user must be notified of the detected threat: EICAR Test File (NOT a Virus!).

An example of a command to test operation of Dr.Web Security Space using the EICAR test program:

```
$ echo 'X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-
FILE!$H+H*' > testfile && drweb-ctl rawscan testfile && rm testfile
```

This command writes the string that represents the body of the EICAR test program to a file named `testfile` created in the current directory, scans the resulting file and removes this file afterwards.

⚠ The abovementioned test requires write access to the current directory. In addition, make sure that it does not contain a file named `testfile` (if necessary, change the file name in the command).

If the test “virus” is detected, the following message is displayed:

```
<path to the current directory>/testfile - infected with EICAR Test File (NOT a
Virus!)
```

If an error occurs during the test, refer to the description of known errors.

If the SpIDer Guard file monitor is enabled, the file can be immediately deleted or quarantined upon detection of the threat (depending on component settings). In this case, after the threat notification is displayed, the `rm` command will inform that the file is missing, which implies that the monitor operates correctly.

7.3. File Monitoring Modes

General Information

The SpIDer Guard file system monitor, which controls access to files, may use three monitoring modes:

- *Regular* (set by default)—monitor file access operations (creating, opening, closing and starting). Scanning of a file, access to which has been allowed, is requested. If a threat is detected upon the scan, an action to neutralize the threat can be applied to the file. Applications are allowed to access the file until the file scanning is finished.
- *Enhanced control of executable files*—monitor files considered non-executable as in regular mode. SpIDer Guard blocks access to an executable file until its scanning is finished.

Executable files are binary files of PE and ELF formats, as well as script files containing the `#!` preamble.

- *"Paranoid" mode*—SpIDer Guard blocks access to any file until its scanning is finished.

Scanner stores file scan results in a specialized cache for a certain time, so reaccessing the same file does not lead to rescanning it if there is information in the cache; the result extracted from the cache is therefore used as the scan result. Despite this, the use of the "paranoid" monitoring mode leads to a significant slowdown in accessing files.

Switching Between File Monitoring Modes

The modes for enhanced monitoring of files and pre-blocking are only available if SpIDer Guard operates in `FANOTIFY` mode and the OS kernel is built with the `CONFIG_FANOTIFY_ACCESS_PERMISSIONS` option enabled.

Switching between the SpIDer Guard monitoring modes is performed using the `cfset` command of the `drweb-ctl` utility.

To switch between SpIDer Guard monitoring modes, superuser privileges are required. To obtain them, you can use the `su` command to switch to a different user or the `sudo` command to perform the action as a different user.

- To switch SpIDer Guard to the `FANOTIFY` mode, use the command:

```
# drweb-ctl cfset LinuxSpider.Mode FANOTIFY
```

- To change the monitoring mode, use the command:

  ```
  # drweb-ctl cfset LinuxSpider.BlockBeforeScan <mode>
  ```

  where <*mode*> defines the blocking mode:

  - `Off`—access is not blocked, SpIDer Guard operates in regular (non-blocking) monitoring mode.
  - `Executables`—access to executable files is blocked, SpIDer Guard performs enhanced monitoring of executable files.
  - `All`—access to all files is blocked, SpIDer Guard monitors files in “paranoid” mode.

- To change an interval within which scan results cached by Scanner remain relevant, use the command:

  ```
  # drweb-ctl cfset FileCheck.RescanInterval <interval>
  ```

  where <*interval*> determines the interval during which cached scan results remain relevant. The acceptable value is from `0s` to `1m`. If you set the interval of less than 1 second, files are scanned upon any request.

8. Working with Dr.Web Security Space

A user can interact with Dr.Web Security Space both in graphical mode via the component that provides a graphical management interface and from the command line (including operation via terminal emulators in graphical mode).

- To start the graphical management interface of Dr.Web Security Space, select the **Dr.Web Security Space** item on the **Applications** or run the following command in a terminal emulator:

  ```
  $ drweb-gui &
  ```

  After that, if a desktop environment is available, the graphical management interface of Dr.Web Security Space is started. To run scanning upon starting the graphical interface or to start its autonomous copy, you can use this command with arguments.
- For details on how to manage Dr.Web Security Space, refer to the Dr.Web Ctl section.
- For graphic desktop environments, scanning can be started from a taskbar (such as Unity Launcher on the Ubuntu OS) and from a graphical file manager (such as Nautilus). Furthermore, the application status indicator appears in the notification area of the desktop, displays pop-up notifications and provides access to the application menu. The indicator is displayed as the notification agent, which, as well as all other service components, starts automatically and its operation does not require intervention. For details, refer to the Integration with Desktop Environment section.
- For details on how to enable an enhanced file monitoring mode in SpIDer Guard, refer to the File Monitoring Modes section.

> Regardless of the selected method to install Dr.Web Security Space, after the installation you need either to activate the license or install a key file if it has already been obtained, or connect Dr.Web Security Space to a centralized protection server (refer to the Registration and Activation section). Until you do that, *anti-virus protection is disabled*.
>
> ---
>
> The mail protocol, IMAP, that is mostly used by mail clients (such as Mozilla Thunderbird) to receive email messages from an email server, works in sessions. Therefore, after adjusting operation of the SpIDer Gate monitor (enabling a previously disabled monitor, changing the mode of scanning of secure connections), it is necessary to restart the email client, so that SpIDer Gate could scan incoming email messages after adjusting its operation mode.

8.1. Operating in Graphical Mode

In this section:

- General Information.
- Notification Agent.
- Graphical Management Interface.

General Information

Two components are responsible for Dr.Web Security Space operation in the desktop environment:

- Notification agent—a component which is automatically launched when the user session starts in the desktop environment. This component displays pop-up notifications on events in the Dr.Web Security Space operation, and also provides a status indicator of Dr.Web Security Space in the area of system notifications and the main menu for interaction with it.
- Graphical management interface—a component that operates in the graphical desktop environment and provides a window interface to manage Dr.Web Security Space operation.

Notification Agent

The Dr.Web Security Space notification agent is designed to:

- display the status indicator of Dr.Web Security Space;
- manage monitors and updating, start the graphical management interface;
- display pop-up notifications about events;
- start scanning according to a specified schedule.

Graphical Management Interface

The graphical management interface of Dr.Web Security Space allows to solve the following tasks:

1. View the status of Dr.Web Security Space operation, including relevance of virus databases and a period of license validity.
2. Start and stop the SpIDer Guard file system monitor.
3. Start and stop the SpIDer Gate network connection monitor.
4. Start on-demand file scanning, including:
   - *Express scan* of system files and most vulnerable system objects.
   - *Full scan* of all system files.
   - *Custom scan* of only specified files and directories or special objects (boot records and active processes).

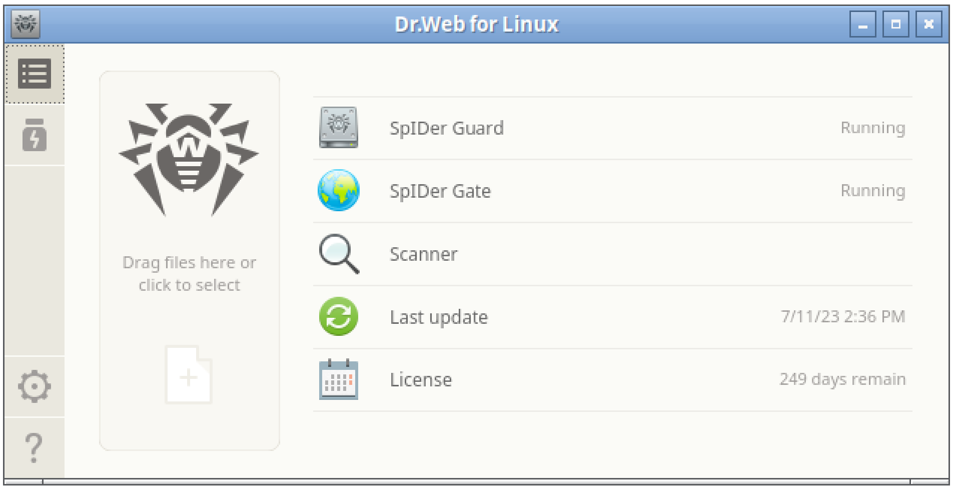
You can select the files to be scanned by specifying target directories or files before scanning and by dragging and dropping them with the mouse from the window of a file manager to the main page (see below) or to the **Scanner** page of the Dr.Web Security Space window.

5. View all threats detected by Dr.Web Security Space during its current session in the graphical mode, including viewing neutralized and skipped threats and quarantined objects.
6. View quarantined objects with a possibility of deleting or restoring them.
7. Configuration of operation parameters of the Dr.Web Security Space components, including the following parameters:
   - Actions that the Scanner and SpIDer Guard automatically apply to the detected threats (according to their type).
   - List of directories and files that are not scanned by the Scanner and are not controlled by the SpIDer Guard file system monitor.
   - Black and white lists of websites and unwanted categories of resources used by the SpIDer Gate monitor, as well as scanning parameters for the files downloaded from the internet or received via email.
   - Scheduled file system scanning, including its frequency and type, and a list of objects for custom scan according to a set schedule.
   - Operation mode (connecting to a centralized protection server or disconnecting from it).
   - Network activity monitoring parameters, including the analysis of encrypted traffic.
   - Permission to use the Dr.Web Cloud service.
8. License management (performed using License Manager).
9. Viewing messages on the anti-virus network status that are sent by the centralized protection server (only if Dr.Web Security Space operates in the anti-virus network and the anti-virus network administrator enables the corresponding setting on the centralized protection server).

⚠ The correct operation of Dr.Web Security Space requires that its service components are started in advance; otherwise, it finishes immediately after startup with the corresponding warning. In a normal operation mode, all necessary service components are started automatically and do not require user intervention.

Appearance of the Graphical Management Interface

Appearance of the Dr.Web Security Space main window of the graphical management interface is shown in the figure below.



Dr.Web for Linux

Drag files here or click to select

SpIDer Guard Running

SpIDer Gate Running

Scanner

Last update 7/11/23 2:36 PM

License 249 days remain

Figure 5. Dr.Web Security Space graphical management interface

The navigation panel is located in the left part of the window. The buttons of the navigation panel allow to perform the actions described below.

Button	Description
1. Continuously Enabled	
☰	Opens the main page where you can: • enable or disable the SpIDer Guard file system monitor; • enable or disable the SpIDer Gate network connection monitor; • start scanning of file system objects (files and boot records) and running processes; • check whether the virus databases are up-to-date and update them, if necessary; • start the License Manager to check the status of the current license and register a new one, if necessary.
⚡	Opens the quarantine page, where you can view quarantined files and delete or restore them.
⚙	Opens Dr.Web Security Space settings window, in particular, of: • scanner of file system objects; • SpIDer Guard file system monitor; • SpIDer Gate network connection monitor; • scheduled scanning. In addition, you can configure the settings of the centralized protection mode.

Button	Description
?	Provides access to reference materials and supportive Doctor Web resources: - product information; - user manual; - Dr.Web forum; - technical support; - **My Dr.Web** personal user webpage. All links are opened in a browser installed on your system.
2. Conditionally Visible	
▮◘▮	Opens the page of the scanning task list, where you can find incomplete (running) scanning tasks. *It is displayed on the navigation panel only if at least one task is running.*
✓ ϟ !	Opens the page with the list of completed scans. The button changes its color depending on the scanning results: 1) green—all scan tasks have been completed successfully; threats were not detected, or all detected threats have been neutralized; 2) red—some of the detected threats have not been neutralized; 3) yellow—at least one scanning task has failed. *It is displayed in the navigation pane only if at least one scanning task was started.*
ϟ	Opens the page with threats detected by the Scanner or by the SpIDer Guard file system monitor. *It is displayed on the navigation panel only if at least one threat was detected.*
🔍	It is displayed on the navigation panel only if the scanning start page is open and active. *When you go to any other page of the main window or scanning is started, the page to start scanning closes automatically, and the button is removed from the navigation panel.*
🖥	It is displayed on the navigation panel only if the SpIDer Guard control page is open and active. *When you go to any other page of the main window, the SpIDer Guard control page closes automatically, and the button is removed from the navigation panel.*
🌐	It is displayed on the navigation panel only if the SpIDer Gate control page is open and active. *When you go to any other page of the main window, the SpIDer Gate control page closes automatically, and the button is removed from the navigation panel.*

Button	Description
🔄	It is displayed on the navigation panel only if the update control page is open and active. *When you go to any other page of the main window, the update control page closes automatically, and the button is removed from the navigation pane.*
📅	It is displayed on the navigation panel only if the License Manager control page is open and active. *When you go to any other page of the main window, the License Manager control page closes automatically, and the button is removed from the navigation panel.*
💬	Opens the message view page from the centralized protection server. *It is displayed on the navigation panel only if Dr.Web Security Space operates in the centralized protection mode and the anti-virus network administrator enables message sending to this workstation.*

Main Page

On the main page of the Dr.Web Security Space graphical management interface, you can see the target pane where you can drag and drop files and directories to be scanned. The pane is marked with the **Drag files here or click to select** label. After objects are dragged and dropped from a file manager window to the Dr.Web Security Space main page, their custom scanning starts (if the Scanner is already scanning other objects, the new task of scanning specified files is queued).

Moreover, the main page of the window has the following buttons:

- **SpIDer Guard**—displays the current state of the SpIDer Guard file system monitor. Click the button to open the control page, where you can start or stop SpIDer Guard and see its operation statistics.
- **SpIDer Gate**—displays the current state of the SpIDer Gate file system monitor. Click the button to open the control page, where you can start or stop SpIDer Gate and see its operation statistics.
- **Scanner**—allows to open the page where you can start scanning files, directories, and other objects of the file system (for example, boot records).
- **Last update**—displays the current status of virus databases. Click the button to open the update control page, where you can start an updating process on demand.
- **License**—displays the status of the current license. Click this button to open the License Manager page, where you can find more detailed information on the current license as well as purchase and register a new license (if required).

8.1.1. Integration with Desktop Environment

Dr.Web Security Space supports the following four methods of integration with a graphical desktop environment:

- displaying the application status icon in the desktop notification area thereby allowing you to open the application context menu;
- calling the context menu with basic file scanning commands when the user right-clicks the application icon in the taskbar;
- starting scanning selected files and directories using the context menu command of a graphical file manager;
- starting scanning files and directories when the user drags and drops them on the main window of Dr.Web Security Space.

Status Indicator in the Notification Area

After the user logs on, the notification agent displays the indicator in the form of the Dr.Web Security Space logo icon in the desktop notification area (if it is supported by your graphical environment). The indicator displays the application status and provides access to the Dr.Web Security Space menu. If any issue occurs (for example, the virus databases are outdated or the license is about to expire), the indicator displays an exclamation mark over the Dr.Web Security Space logo: ⚠.

In addition to the status indicator, the notification agent also displays pop-up notifications that inform the user of important events of Dr.Web Security Space operation, such as:

- detected threats (including those detected by the SpIDer Guard and SpIDer Gate real-time monitors);
- license validity period is about to expire.

If you click this icon, the Dr.Web Security Space context menu opens:

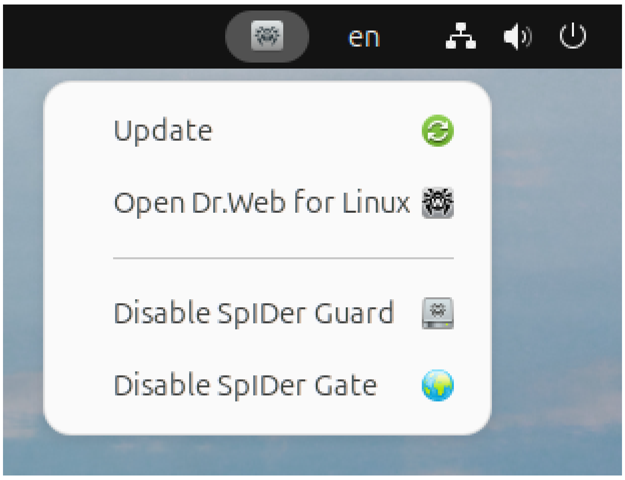


Figure 6. Dr.Web Security Space indicator context menu

When you select the **Open Dr.Web Security Space** item, the graphical interface management window of Dr.Web Security Space appears on the screen; that is, the application is starting. Selection of the **Enable SpIDer Gate** or **Disable SpIDer Gate** and **Enable SpIDer Guard** or **Disable SpIDer Guard** menu items starts or stops operation of the corresponding monitor.

> You need to authenticate as a user with administrative privileges to disable any monitor (refer to Managing Application Privileges).

Selection of the **Update** item forces an update procedure to start.

If the indicator notifies of issues in Dr.Web Security Space operation, the icon of the component that caused the issue is also provided with an exclamation mark, for example: ⚠.

Status Indicator Issues

1. If the indicator displays a critical error mark ✖ and the drop-down menu contains only the disabled **Loading** item, then Dr.Web Security Space cannot start because some service components are unavailable. If this status is permanent, try to resolve this error manually or contact our technical support.
2. If the indicator is not displayed in the notification area after the user logged in, try to resolve this error manually or contact our technical support.

> In different desktop environments, appearance and behavior of the indicator can differ from the ones described above; for example, icons may not be displayed on the drop-down menu.

Context Menu on Taskbar Icon

The application can be opened by selecting the **Dr.Web Security Space** item of the **Applications** menu. If the desktop environment contains a taskbar, the button with the application icon appears on the taskbar when Dr.Web Security Space is started. Right-clicking the button opens the application menu which may look as follows (the Unity Launcher menu on the Ubuntu OS):

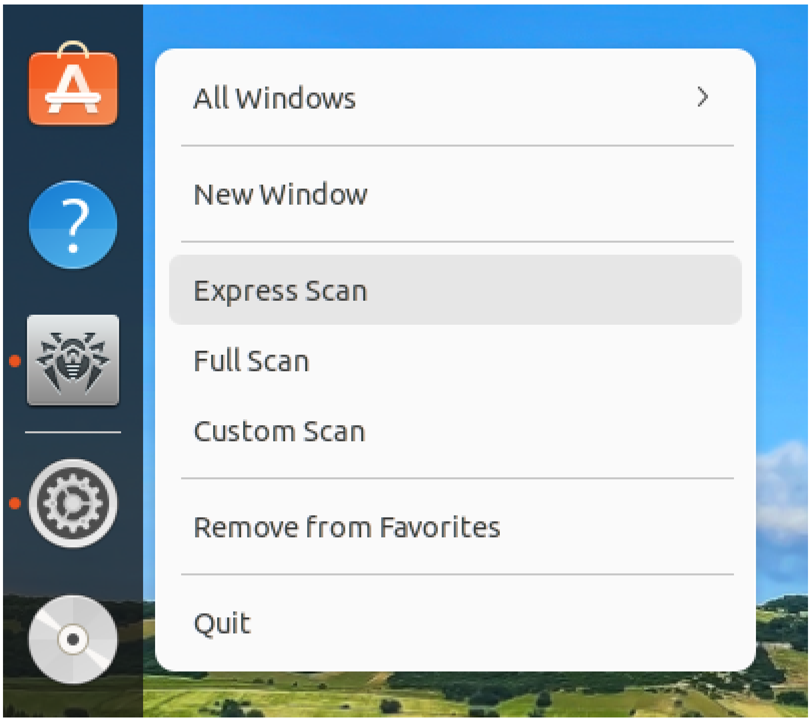


Figure 7. The Dr.Web Security Space context menu on the taskbar

- Selection of the **Express scan** items, **Full scan** items and the **Custom scan** items allows you to start the corresponding scan task (for the **Custom scan** item it opens the page where you can select objects to be scanned).
- Selection of the **Dr.Web Security Space** menu item starts the graphical interface (if not started), while selection of the **Quit** item terminates it (if currently running).
- Selection of the **Lock to Launcher** item allows you to lock the application icon on the taskbar to quickly start the graphical interface and basic scan tasks.

In case there are running tasks for file system scanning in the task queue, the indicator of the total execution of the active scanning tasks is displayed on the top of the application icon in the taskbar.

In different desktop environments, the taskbar as well as the context menu and behavior of the menu items (excluding **Express scan**, **Full scan** and **Custom scan**) may differ from the ones described above.

Taskbar Icon Issues

If the button with the application icon is displayed on the taskbar but the context menu does not contain items for starting of scan tasks, try to open the application via the **Dr.Web Security Space** item on **Applications** menu (instead of opening the application using the `drweb-gui` command in a terminal emulator or selecting the **Open Dr.Web Security Space** item in the context menu of the status indicator in the notification area).

Starting Scan From File Manager Context Menu

Dr.Web Security Space allows you to scan files and directories directly from the window of a graphical file manager, such as Nautilus. To scan the files and directories:

1. Select them in the file manager window and then click right mouse button.
2. Select the **Open With Other Application** item in the appeared context menu.
3. Find and select Dr.Web Security Space item on the list of installed applications.

Usually, after the first use of Dr.Web Security Space for opening files, this association will be saved by the file manager, and the **Open With Dr.Web Security Space** item will be further available in the context menu.

> In different graphic file managers, the item of the context menu as well as the way to choose an application for processing the selected files may differ from the ones described above.

Issues with Using the Context Menu of the File Manager

If Dr.Web Security Space is selected in a file manager using the **Open With Other Application** context menu item, some graphical environments for GNU/Linux can automatically create associations for files or directories (based on their MIME values) with Dr.Web Security Space. In this case, opening these files and directories in the file manager runs Dr.Web Security Space. To resolve this issue, remove the created association.

Drag and Drop Files and Directories to a Window of the Graphical Management Interface

Dr.Web Security Space allows you to start scanning of files and directories when you drag them from a file manager window using the mouse pointer and drop them on the Dr.Web Security Space window. To start scanning of dragged and dropped files and directories, it is necessary for the main page or page with scan types of the window to be opened. If a page of the Dr.Web Security Space window contains an area with the label **Drag files here or click to select**, this page supports dragging and dropping files and directories to be scanned.

8.1.2. Starting and Closing

Starting Dr.Web Security Space Graphical Management Interface

To start the Dr.Web Security Space graphical management interface:

- select the **Dr.Web Security Space** item in the **Applications** system menu or

- right-click the Dr.Web Security Space indicator icon in the notification area and select **Open Dr.Web Security Space** from the drop-down list.

You can also start the Dr.Web Security Space graphical management interface from the command line by running the `drweb-gui&` command in a terminal emulator.

Closing Dr.Web Security Space Graphical Management Interface

To close the Dr.Web Security Space graphical management interface, close the window using the standard close button on the title bar.

> Service components, including the notification agent and the SpIDer Guard and SpIDer Gate monitors, continue their operation after closing the Dr.Web Security Space graphical interface (unless they are disabled by the user).
>
> Under normal operation, all necessary service components do not require user intervention.

8.1.3. Threat Detection and Neutralization

Threat detection and neutralization can be performed either by Scanner (on user demand or on schedule) or by the SpIDer Guard file system monitor and SpIDer Gate network connection monitor.

- To enable or disable SpIDer Guard and SpIDer Gate, use the context menu in the notification area or open the corresponding page with the monitor settings (refer to File System Monitoring and Monitoring Network Connections).
- To view and manage Scanner current tasks for scanning file system objects, open the task management page.
- To view threats detected by Scanner or the SpIDer Guard file system monitor, open the page with listed threats.
- To manage quarantined threats, open the quarantine management page.
- To configure the reaction of Dr.Web Security Space to detected threats, open the settings window. On this window, you can also enable and adjust a schedule of periodic scanning as well as configure monitoring of encrypted connections.

> If Dr.Web Security Space is contolled by a centralized protection server that does not allow the user to start scanning, the **Scanner** page of the Dr.Web Security Space window is disabled. Furthermore, in this case the notification agent and the graphical management interface do not start scheduled scanning.

8.1.3.1. Scanning on Demand

In this section:

- Scan Types.
- Starting Scanning.
- Editing the List of Custom Scan Objects.
- Starting a Custom Scan of Listed Objects.

Scan Types

Scanner can start the following scan types on user demand:

- *Express scan*—scan a strictly defined set of critical system objects that are at high risk to be compromised (boot records, system files and so on).
- *Full scan*—scan all local file system objects available to the user under whom Dr.Web Security Space is started.
- *Custom scan*—scan file system objects or some special-type objects directly specified by the user.

> If Dr.Web Security Space operates in centralized protection mode and the centralized protection server does not allow the user to start file scanning, this page of the Dr.Web Security Space window is disabled.
>
> Scanning increases processor load, which can cause the battery to discharge faster in case of running on mobile devices. Thus, it is recommended to perform a scan of a portable computer when it is plugged in.

Starting Scanning

To start scanning of file system objects, click **Scanner** on the main page of the window.

The page with scan types opens. To start an *Express* or *Full* scan, click the corresponding button. After that, the scanning starts automatically.

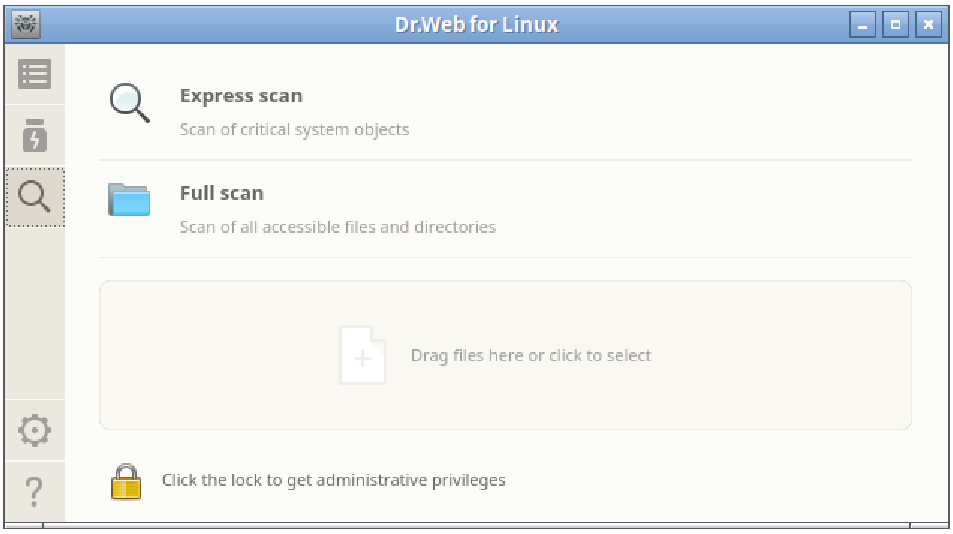

Dr.Web for Linux
Express scan
Scan of critical system objects
Full scan
Scan of all accessible files and directories
Drag files here or click to select
Click the lock to get administrative privileges

Figure 8. Selection of a scan type

Scanning of objects is always performed by Scanner with current application privileges. If the application does not have elevated privileges, all files and directories that are inaccessible to the user who started Dr.Web Security Space will be skipped. To enable scanning of all required files that are not owned by you, elevate the application privileges before scanning (refer to the Managing Application Privileges section).

To start the *Custom scan* of only required files and directories, do one of the following:

- **Drag and drop**

  Open the page for selecting a scan type, drag the required files and directories using the mouse from a file manager window and drop them in the area marked with the **Drag files here or click to select** label. You can also drag the objects and drop them in the main page of the Dr.Web Security Space window.

  When dragging files and directories to the window, it displays an area with the **Drop files here** label. To start scanning of the selected files, drop them in the page by releasing the mouse button. After that, the scanning will start automatically.

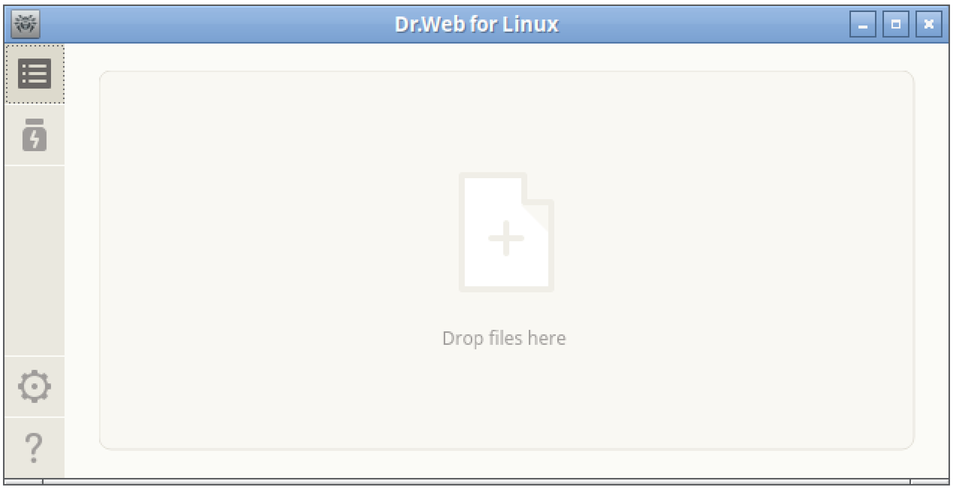
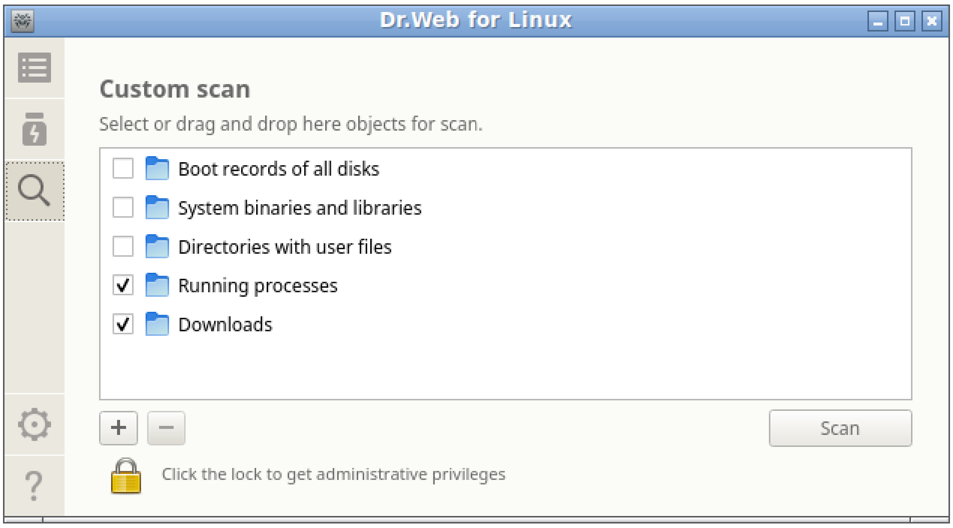


Figure 9. Area for adding files

- **Adding objects for the custom scan**

  To select objects for the custom scan, click the area for adding files. A list of objects for the custom scan will be displayed on the screen.



Custom scan

Select or drag and drop here objects for scan.

☐ Boot records of all disks
☐ System binaries and libraries
☐ Directories with user files
☑ Running processes
☑ Downloads

Click the lock to get administrative privileges

Figure 10. List of objects for the custom scan

The list contains four special items denoting predefined groups of objects:

- *Boot records of all disks*—all boot records of all available disks.
- *System binaries and libraries*—all directories with system executable files (`/bin`, `/sbin` and so on).
- *Directories with user files*—directories with user files and files of the current session (the `/home/`<*username*> home directory (`~`), `/tmp`, `/var/mail`, `/var/tmp`).
- *Running processes*—executable files that started currently running processes. At that, if a threat is detected in an executable file, all processes started with it are forcibly terminated and the threat is neutralized.

Editing the List of Custom Scan Objects

If required, you can add custom paths to the list of the objects to be scanned. For that purpose, drag and drop required objects (paths to the objects are automatically added to the list) or click + below the list. After that, a standard dialog window opens, where you can select files and directories. Select a required object (a file or a directory) and click **Open**.

> Hidden files and directories are not displayed in the file chooser by default. To view such objects, click ✱ on the toolbar of the file chooser.

Click − below the list to remove all selected paths from the list (a path is selected if the list item containing this path is selected). To choose several paths, select items in the list and hold SHIFT or CTRL. Please note that the first four items in the list are predefined and cannot be removed.

Starting a Custom Scan of Listed Objects

To start a custom scan, select all check boxes for the objects to be scanned and click **Scan**. After that, the scanning starts.

After the scanning starts, the new task is added to the queue which contains all Scanner tasks of the current session: complete tasks, tasks in progress and pending tasks. You can view the list of tasks and manage them on the scan task management page.

8.1.3.2. Scheduled Object Scanning

Dr.Web Security Space can perform the automatic launch of scheduled scanning of the specified list of the file system objects according to the indicated schedule.

> If Dr.Web Security Space is operating under the server control in the centralized protection mode and launch of scanning on demand is prohibited on the centralized protection server, this Dr.Web Security Space option is unavailable.

Scanning Types

According to schedule, it is possible to perform the following types of scanning:

- *Express scan*—scan of critical system objects that are at high risk to be compromised (boot records, system files, and so on).
- *Full scan*—scan of all file system objects available for the user under whom Dr.Web Security Space is started.
- *Custom scan*—scan of file system objects or other special objects specified by the user.

Starting Scanning

Scanning is started automatically according to the set schedule. Start of the scanning is performed by:

1. The graphical interface itself if it runs when the scanning starts.
2. The notification agent if the graphical interface in unavailable when the scanning starts.

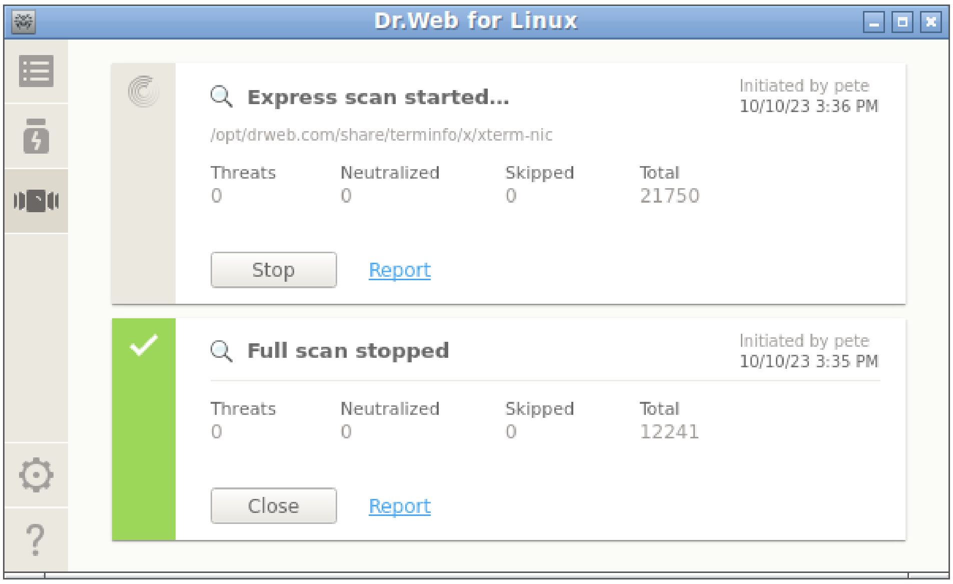
When scheduled scanning starts, the graphical interface for management automatically starts (if it is not launched yet), the created task is added to the queue which contains all scanning tasks of the current session: complete tasks, tasks in progress, and pending tasks. You can view the list of tasks and manage them on the scan task management page.

8.1.3.3. Managing Scan Tasks

You can view the list of created tasks and tasks in progress on the special Dr.Web Security Space page. If at least one task is queued, a button that opens the page with the task list becomes visible in the navigation pane. Depending on the status of the queued tasks, the button has one of the following icons:

◉	At least one of the tasks is not complete (an animation is used).
✓	All scanning tasks in the list are complete or stopped by the user; no threat is detected or all detected threats are successfully neutralized.
⚡	All scanning tasks in the list are complete or stopped by the user; some of the detected threats are not neutralized.
!	All scanning tasks in the list are complete or stopped by the user. Some of the tasks have failed.

Tasks are sorted by creation time (from the last to the first created task).



Express scan started...

Initiated by pete
10/10/23 3:36 PM

/opt/drweb.com/share/terminfo/x/xterm-nic

Threats	Neutralized	Skipped	Total
0	0	0	21750

Full scan stopped

Initiated by pete
10/10/23 3:35 PM

Threats	Neutralized	Skipped	Total
0	0	0	12241

Figure 11. Task list

For each listed task, the following information is available:

- scanning type (*Express scan*, *Full scan*, *Custom scan*, or Scheduled scan);
- name of the user who started scanning (if unknown, the system identifier of the user (*UID*) is displayed);
- date of task creation and completion (if complete);
- number of detected threats, neutralized threats, skipped files, and total number of scanned objects.

The status of the task is indicated with the color mark assigned to the listed task. The following colors are used:

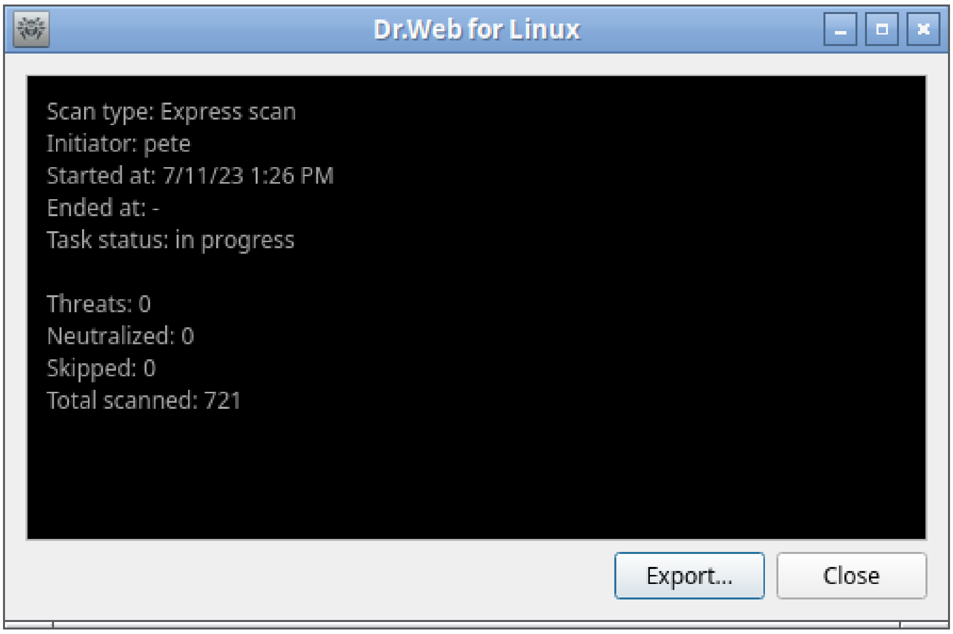
▮	Scanning is not complete or is pending.
🟩	Scanning is complete or stopped by the user; no threat is detected or all detected threats are neutralized.
🟧	Scanning is stopped due to an error.
🟥	Scanning is complete or stopped by the user; at least one detected threat is not neutralized.

> The list contains only those scanning tasks performed by Scanner that were directly created by the user in the Dr.Web Security Space window, as well as scanning tasks that were started automatically according to the set schedule.

On the task description area, one of the following buttons is available:

- **Cancel**—cancel the pending task. The button is available if the task is pending. Once the button is clicked, the task completes. Information on the task remains in the list.
- **Stop**—stop the task which is in progress. After you click this button, the stopped task cannot be resumed. The button is available if the task is in progress. Information on the stopped task remains in the list.
- **Close**—close information on the complete task and delete the task from the list. The button is available if the task is not complete and if all detected threats are neutralized.
- **Neutralize**—neutralize threats. The button is available if the task is complete and some of the detected threats are not neutralized.
- **Details**—open the list with detected threats and neutralize them. The button is available if the task is complete and some of the detected threats are not neutralized.

Click the **Report** link to open a report window with scanning results including the detailed information about the task and the list of detected threats, if any.



```
Scan type: Express scan
Initiator: pete
Started at: 7/11/23 1:26 PM
Ended at: -
Task status: in progress

Threats: 0
Neutralized: 0
Skipped: 0
Total scanned: 721
```

Figure 12. Detailed information on scanning results

File systems of UNIX-like operating systems, such as GNU/Linux, can contain special objects that appear as named files but are not actual files containing data (for example, such objects are symbolic links, sockets, named pipes, and device files). They are called *special* files as opposed to *usual* (*regular*) ones. Dr.Web Security Space *always* skips special files during scanning.

If you click a link with the detected threat name, a default web browser opens a page with the information about the threat (the Doctor Web official website is displayed; an internet connection is required).

Click **Export** if you want to save the scanning report to a text file. To close the window with detailed scanning information, click **Close**.

To any threat detected by the Scanner during scanning of any type started in a graphical mode (including a scheduled scanning), Dr.Web Security Space applies actions that are specified in the settings on the **Scanner** tab.

Threat neutralization settings specified on the **Scanner** tab are not used for *centralized scanning*.

To view all detected threats, open the page with listed detected threats.

8.1.3.4. File System Monitoring

In this section:

- General Information
- Managing Operation of File System Monitor

- Configuring File System Monitor
- Issues with SpIDer Guard Operation

General Information

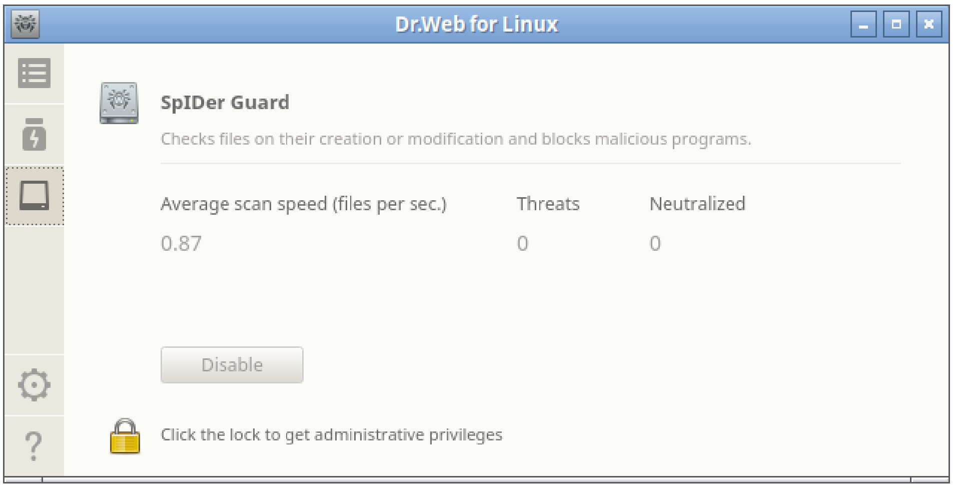
File system objects are continuously controlled by the SpIDer Guard file system monitor.

The Dr.Web Security Space graphical management interface allows you to configure the operation of SpIDer Guard, in particular to:

- start and stop the file system monitor;
- view component statistics and the list of detected threats;
- configure the following parameters of the file system monitor:
  - reaction to detected threats;
  - list of scanning exclusions.

Managing Operation of File System Monitor

You can start and stop the SpIDer Guard file system monitor and view statistics on its operation on a special page of Dr.Web Security Space. To access the page, click **SpIDer Guard** on the main page.



Checks files on their creation or modification and blocks malicious programs.

Average scan speed (files per sec.)	Threats	Neutralized
0.87	0	0

Click the lock to get administrative privileges

Figure 13. SpIDer Guard operation management

The following information is displayed on the page for monitoring management:

- status of the SpIDer Guard file system monitor (enabled or disabled) and details on an error if it occurred during component operation;
- file system monitoring statistics:
  - average file scanning speed;
  - number of detected and neutralized threats.

To enable monitoring, if disabled, click **Enable**. To disable monitoring, if enabled, click **Disable**.

> To disable the file system monitor, the application must operate with elevated privileges, refer to the Managing Application Privileges section.
>
> ---
>
> The option to enable and disable the SpIDer Guard file system monitor when Dr.Web Security Space is managed by a centralized protection server can be blocked if disabled by the server.

The SpIDer Guard status (enabled or disabled) is shown by the indicator:

	SpIDer Guard file system monitor is enabled and protecting the file system.
⚠	SpIDer Guard file system monitor is not protecting the file system because either the user has disabled the component or an error has occurred.

To close the page, go to another page by using the buttons in the pane.

The list of threats detected by SpIDer Guard in the current Dr.Web Security Space session is displayed on the detected threats view page (available if at least one threat is detected).

Configuring File System Monitor

Configure the SpIDer Guard file system monitor in the settings window:

- on the **SpIDer Guard** tab, specify a reaction to detected threats;
- on the **Exclusions** tab, specify objects to be excluded from monitoring.

> For details on enabling the enhanced file monitoring mode for SpIDer Guard, refer to the File Monitoring Modes section.

Issues with SpIDer Guard Operation

If an error occurs in operation of SpIDer Guard, the management page displays an error message. To fix the error, refer to the description of known errors in Appendix G.

8.1.3.5. Monitoring of Network Connections

In this section

- General Information
- Managing Operation of the Network Connection Monitor
- Configuring SpIDer Gate
- Issues with SpIDer Gate Operation

General Information

Continuous control of established network connections is performed by SpIDer Gate. It restricts access to websites added to user black lists or belonging to categories marked as unwanted for visiting. In addition, SpIDer Gate scans:

- incoming and outgoing email messages, including attachments (among other things, for signs of spam);
- files downloaded from the internet.

If SpIDer Gate detects a threat in the scanned object, its receiving or sending is blocked.

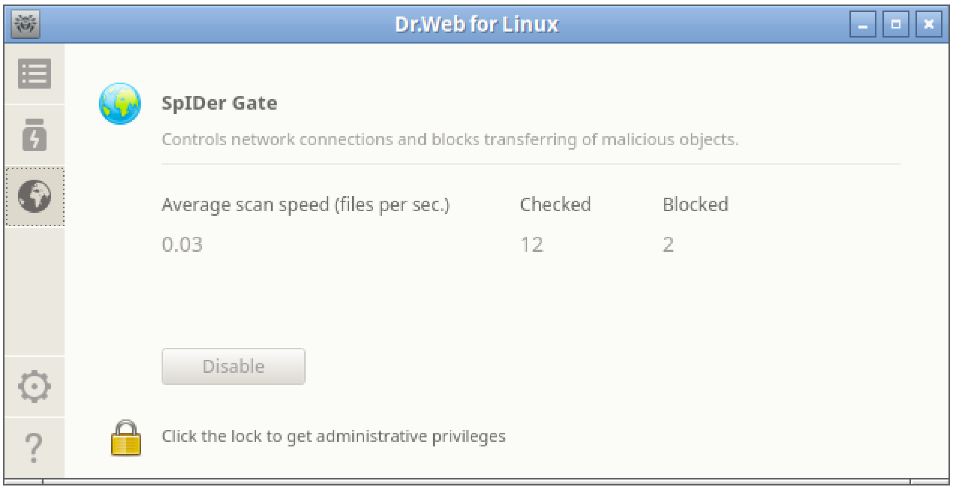
The Dr.Web Security Space graphical management interface allows you to configure the operation of SpIDer Gate:

- start and stop the network connection monitor;
- view the number of scanned and blocked objects and attempts to access websites;
- configure the following parameters of network connection monitoring:
  - type of traffic to be scanned (web traffic, FTP traffic);
  - list of websites and hosts access to which is restricted;
  - personal black and white lists of websites and hosts;
  - parameters of scanning files downloaded from the internet.

Threats in email messages can be detected by the SpIDer Guard file system monitor (if enabled) at the moment of their saving by the mail client to the local file system.

Managing Operation of the Network Connection Monitor

You can start and stop the SpIDer Gate network connection monitor and view statistics on its operation on the specialized page of Dr.Web Security Space. To access this page, click **SpIDer Gate** on the main page.


Dr.Web for Linux

SpIDer Gate

Controls network connections and blocks transferring of malicious objects.

Average scan speed (files per sec.)	Checked	Blocked
0.03	12	2

Click the lock to get administrative privileges

Figure 14. SpIDer Gate management page

On the page for monitoring network connections, the following information is displayed:

- state of the SpIDer Gate network connection monitor (enabled or disabled) and details on errors if they occurred during the component operation;
- monitoring statistics:
  - average speed of scanning email messages and files downloaded from the internet;
  - number of scanned objects (email messages, files downloaded from the internet and URLs);
  - number of blocked attempts to access websites and malicious objects.

To enable monitoring, if disabled, click **Enable**. To disable monitoring, if enabled, click **Disable**.

> To disable monitoring network connections, the application must operate with elevated permissions; refer to the Managing Application Privileges section.
>
> The option to enable and disable the SpIDer Gate network connection monitor when Dr.Web Security Space is managed by a centralized protection server can be blocked if disabled by the server.

State of the SpIDer Gate network connection monitor (enabled or disabled) is indicated as follows:

🌐	SpIDer Gate is enabled and controls network connections (sending and receiving email messages and accessing the internet).
🌐⚠	SpIDer Gate does not control network connections (access to websites is not restricted, email messages being received and sent and downloaded files are not scanned) because either the user disabled the component or an error occurred.

> If a mail client (such as Mozilla Thunderbird) is running that uses the IMAP protocol to receive email messages, it is necessary to restart such a mail client after enabling the SpIDer Gate monitor to ensure scanning of incoming email messages.

To close the page for monitoring network connections, go to another page by using the buttons in the pane.

SpIDer Gate Operation Settings

The SpIDer Gate network connection monitor can be configured in the settings window:

- on the **SpIDer Gate** tab—setting the list of blocked website categories and reaction to the detected threats;
- on the **Exclusions** tab—managing the black and white lists of websites and exclude application network activity from monitoring;
- on the **Network** tab—managing the scan of protected connections (SSL/TLS).

Issues with SpIDer Gate Operation

If an error occurs in operation of the network connection monitor, the management page displays the error message. To solve the issue, refer to the description of known errors in the Appendix G. Known Errors section.

> Depending on the distribution, Dr.Web Anti-Spam can be not bundled with Dr.Web Security Space. In this case, email messages are not scanned for signs of spam.
>
> ---
>
> If any email messages are falsely detected by the Dr.Web Anti-Spam component, we recommend you to forward them to special addresses for analysis and improvement of spam filter quality. To do that, save each message to a separate `.eml` file. Attach the saved files to an email message and forward it to the corresponding service address:
>
> - nonspam@drweb.com—if it contains email files *erroneously recognized as spam*;
> - spam@drweb.com—if it contains spam email files *failed to be recognized as spam*.

8.1.3.6. Viewing Detected Threats

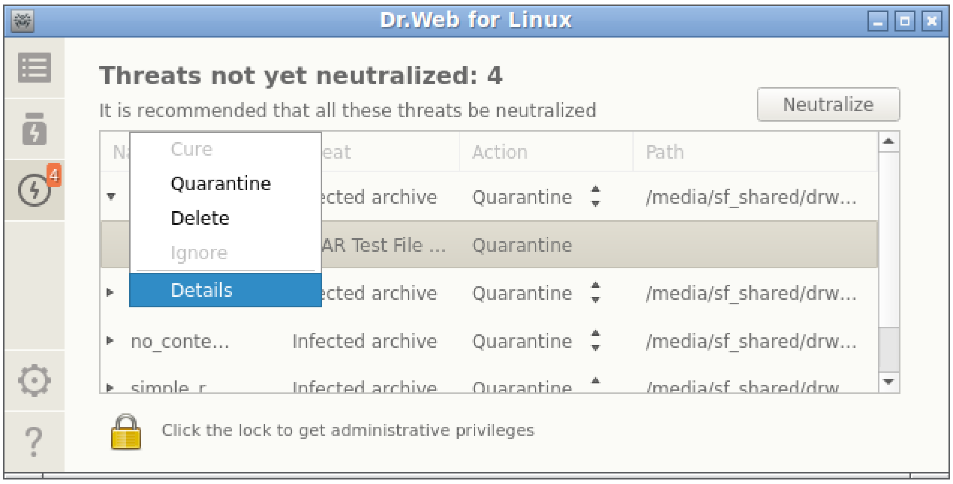
In this section:

- General Information.
- Neutralizing Detected Threats.
- Viewing Threat Details.

General Information

The list of threats detected by Scanner and the SpIDer Guard file system monitor during the current Dr.Web Security Space session is displayed on a special window page which is available only if at least one threat was detected.

If threats were detected, click ⚡ in the navigation pane to open a page with a list of threats.



Threats not yet neutralized: 4

It is recommended that all these threats be neutralized

Na	eat	Action	Path
▾	cted archive	Quarantine	/media/sf_shared/drw...
	AR Test File ...	Quarantine	
▸	cted archive	Quarantine	/media/sf_shared/drw...
▸ no_conte...	Infected archive	Quarantine	/media/sf_shared/drw...
▸ simple_r...	Infected archive	Quarantine	/media/sf_shared/drw...

Click the lock to get administrative privileges

Figure 15. Viewing threats

The list provides the following information about each detected threat:

- a name of an object containing a threat;
- a name of the threat contained in the object as classified by the Doctor Web company;
- an action to be applied to the object for neutralizing the threat (or an action that was already applied, if the threat was neutralized);
- a path to the file system object in which this threat was detected.

Neutralized threats are displayed in the list as grayed out items.

Neutralizing Detected Threats

If some of the listed threats are not neutralized, the **Neutralize** button becomes available above the list. Once the button is clicked, an action specified in the corresponding **Action** field is applied to each threat that was not neutralized. If the threat is successfully neutralized, its row in the table becomes grayed out. If an attempt to neutralize the threat fails, the listed item is displayed in red and an error message appears in the **Action** field.

By default, an action to be applied to a threat is selected according to the settings of the component that detected the threat. You can adjust default actions applied to the threats detected by Scanner and the SpIDer Guard file system monitor on the corresponding tabs of the settings window.

If the *Report* action was selected in Scanner or SpIDer Guard settings for a certain threat type, all threats of this type will be displayed with the *No action* action in the threat list. To neutralize such threats, indicate an action for each of them in the **Action** field.

If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined and not deleted.

If you need to apply an action different from the one specified in the settings, click the **Action** field in the row for the threat and select the required action from the context menu.

You can select several threats in the list at once. To do that, select them with the mouse while holding CTRL or SHIFT:

- When you hold CTRL, threats are selected one by one.
- When you hold SHIFT, threats are selected contiguously.

After selecting threats, you can apply some action to them by right-clicking the selection and then selecting the required action from the drop-down list. The selected action will be applied to all selected threats.

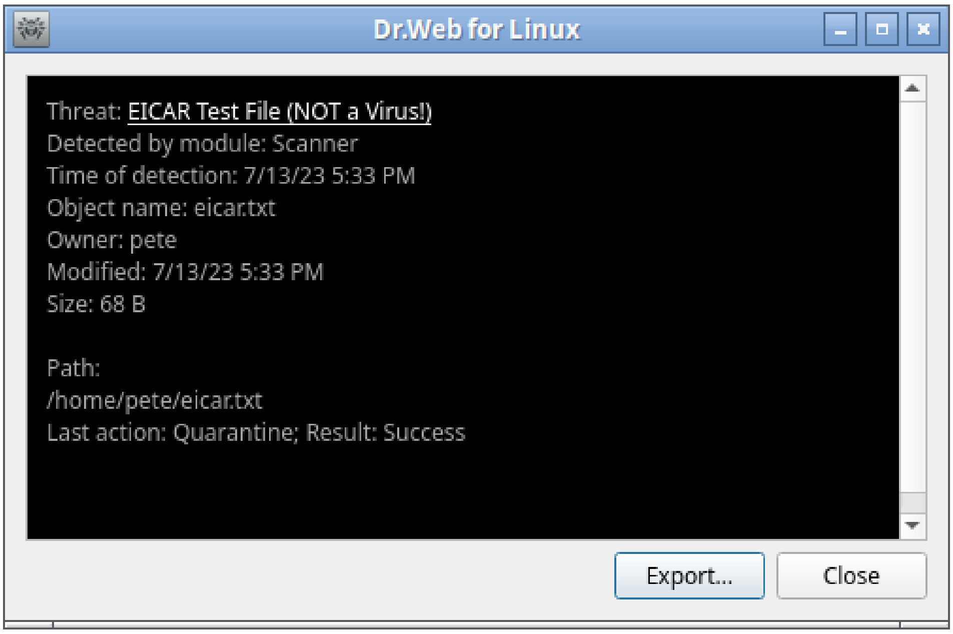
> ⚠ If a threat is detected in a compound object (an archive, an email message and so on), the selected action is applied to the container as a whole and not to the nested infected object.

> The *Cure* action cannot be applied to some threat types.
>
> If necessary, elevate application privileges to enable successful neutralization of threats.

The threats to which the *Ignore* action was applied will be displayed in the list until the graphical management interface is restarted.

Viewing Threat Details

To get the detailed information about any detected threat, right-click the corresponding row and select **Details** in the appeared context menu. This opens the window with the details on the threat and the object containing it. If you need to get details on several threats at once, select them with the mouse from the list while holding CTRL before opening the context menu.



Threat: EICAR Test File (NOT a Virus!)
Detected by module: Scanner
Time of detection: 7/13/23 5:33 PM
Object name: eicar.txt
Owner: pete
Modified: 7/13/23 5:33 PM
Size: 68 B

Path:
/home/pete/eicar.txt
Last action: Quarantine; Result: Success

Figure 16. Threat details

This window displays the following information:

- a threat name as classified by the Doctor Web company;
- the Dr.Web Security Space component that detected the threat;
- the date and time of threat detection;
- the information about the file system object in which the threat was detected: its name, user/owner, last modified date and a path to the object in the file system;
- the last action applied to the threat and the result (if an option to apply actions automatically is enabled in the component settings, for example, you can set it for Scanner on the corresponding tab of the settings window).

Click the name of the threat to open a webpage with its description (the Doctor Web official website will be visited; an internet connection is required).

Click **Export** to save the information displayed in the window to a text file (the file chooser will open). To close the window with threat and object details, click **Close**.

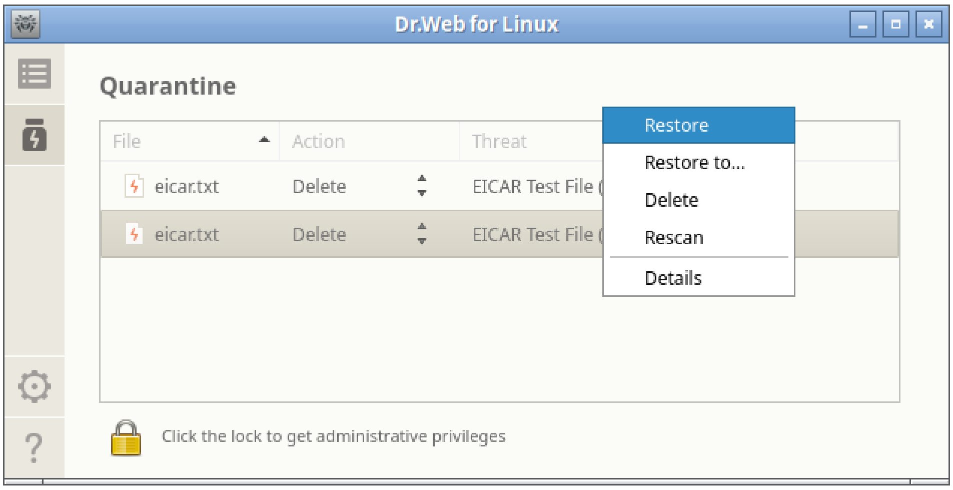
8.1.3.7. Managing Quarantine

In this section:

- General Information.
- Applying Actions to Quarantined Objects.
- Viewing Details on Quarantined Objects.

General Information

The list of objects quarantined by Dr.Web Security Space is displayed on a special page. To open this page, click on the navigation pane.



Quarantine

File	Action	Threat
eicar.txt	Delete	EICAR Test File (
eicar.txt	Delete	EICAR Test File (

Click the lock to get administrative privileges

Figure 17. Quarantine management

If the quarantine is not empty, the following information is displayed for every threat:

- a name of a malicious object;
- an action to be applied to the quarantined object;
- a name of the threat contained in the object as classified by the Doctor Web company.

Applying Actions to Quarantined Objects

To apply an action to a quarantined object, right-click the row with the information about this object and select a required action from the context menu. If you need to apply an action to several quarantined objects, select the corresponding rows in the list before opening the context menu. To select several rows, hold CTRL or SHIFT:

- When you hold CTRL, quarantined objects are selected one by one.
- When you hold SHIFT, quarantined objects are selected contiguously.

The context menu provides for the following actions:

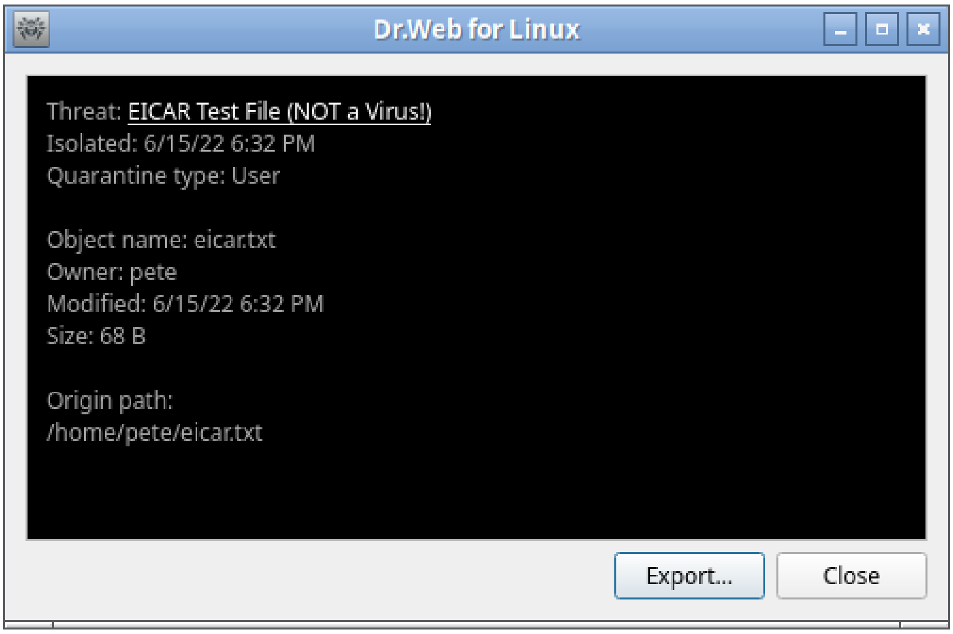
- **Restore**—restore the selected objects to their original location on the file system.
- **Restore to**—restore the selected objects to a specified file system location (a window for choosing a target directory will appear).
- **Delete**—irreversibly delete the selected objects.
- **Rescan**—scan the selected objects again and cure them if possible.

If the selected action is successfully applied to the selected object, its row is removed from the table. If the attempt fails, the corresponding row remains active, its text becomes red and the **Action** field displays details on the error.

> To apply actions to quarantined objects, it may be necessary to elevate application privileges; in particular, this is necessary to apply actions to the objects quarantined by another user.

Viewing Details on Quarantined Objects

To receive detailed information about any quarantined object, right-click the corresponding row and select **Details** from the context menu. This opens a window with the detailed information about the object. If you need to get the detailed information about several quarantined objects, select them in the list before opening the context menu.



Threat: EICAR Test File (NOT a Virus!)
Isolated: 6/15/22 6:32 PM
Quarantine type: User

Object name: eicar.txt
Owner: pete
Modified: 6/15/22 6:32 PM
Size: 68 B

Origin path:
/home/pete/eicar.txt

Figure 18. The information about the quarantined object

This window displays the following information:

- a threat name as classified by the Doctor Web company;
- the date and time when the object was quarantined;
- a type of the quarantine where the object is isolated;
- a name of the last applied action and its result;
- the information about the quarantined file system object: its name, user/owner, last modification date and path on the file system.

Click the name of the threat to open a webpage with its description (the Doctor Web official website will be visited; an internet connection is required).

Click **Export** to save the information displayed in the window to a text file (a file chooser window will open). To close the window with the detailed information about the threat and the object containing it, click **Close**.

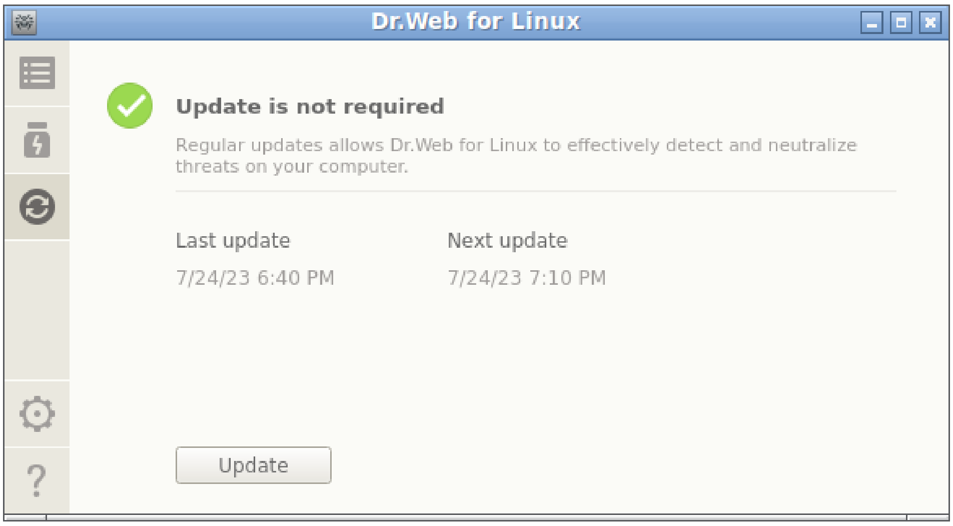
8.1.4. Updating Antivirus Protection

In this section:

- General Information.
- Configuring Updates.
- Updating Offline.
- Issues with Updater.

General Information

Periodic updates of virus and web categories databases as well as the anti-virus engine are downloaded and installed by Updater automatically. You can view the update status and, if required, force updating on a specialized page of the Dr.Web Security Space window. Virus databases are considered outdated after a day from the moment of the last successful update. To open the update management page, click **Last update** on the main page.



Update is not required

Regular updates allows Dr.Web for Linux to effectively detect and neutralize threats on your computer.

Last update: 7/24/23 6:40 PM

Next update: 7/24/23 7:10 PM

Figure 19. Update management

The update management page displays the following information:

- status of the virus databases, databases of web resource categories and of the scanning engine;
- information about the last successful update and the date of the next scheduled update.

To force updating, click **Update**. To close the update management page, select another page by clicking a corresponding button on the navigation pane.

If Dr.Web Security Space operates in a centralized protection mode, this page will be blocked.

Configuring Updates

You can configure Dr.Web Security Space update settings in the settings window on the **Main** tab.

Updating Offline

Updating anti-virus protection offline is available only via the command line. Command examples can be found in the corresponding section.

Issues with Updater

If Updater fails, error information is displayed on the update management page. To resolve the issue, refer to Appendix G, where you can find a detailed description of the known errors.

8.1.5. License Manager

In this section:

- General Information.
- Starting the License Manager.
- License Activation.
- Deleting License Key File.

General Information

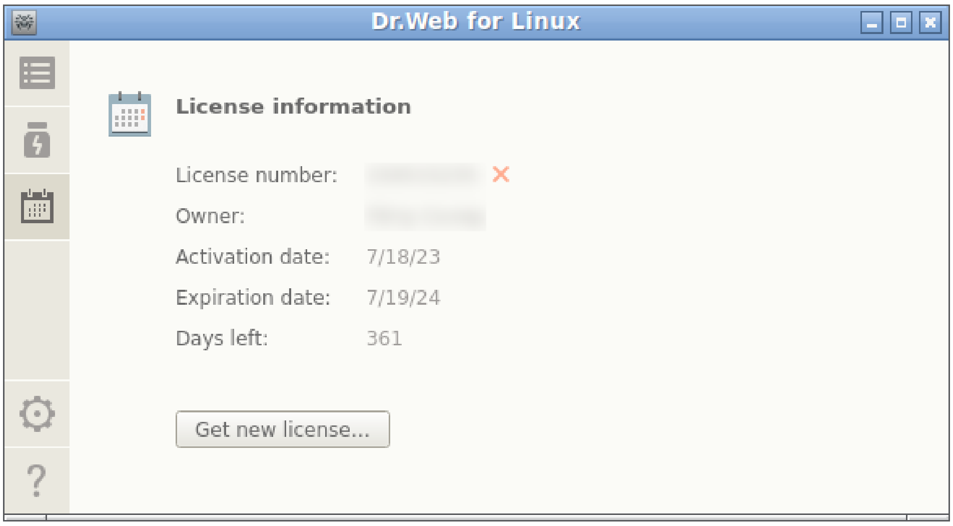
License Manager allows to view information on the current license issued for the user of Dr.Web Security Space. License data is stored in a license key file that provides operation of Dr.Web Security Space on the user computer. If neither license key file nor demo key file is found on the computer, all Dr.Web Security Space functions (including the file scan, file system monitoring, virus database update) are blocked.

Starting the License Manager

License Manager page is available in the Dr.Web Security Space graphical interface. To open the page, click **License** on the main page.

If a key file associated with some license for Dr.Web Security Space granted to the user or with an active demo period is installed, the License Manager start page displays information about the license including its number, owner, and period retrieved from the key file.

The figure below shows appearance of the License Manager page.



License information

License number: [redacted] ✕

Owner: [redacted]

Activation date: 7/18/23

Expiration date: 7/19/24

Days left: 361

Figure 20. License information page

To delete a license key file, click ✕ next to the license number.

The License Manager page will close if you select any other page by clicking a corresponding button on the navigation pane.

License Activation

To activate a license or a demo period via License Manager (in particular, to purchase a new license or renew the current one) and obtain the corresponding key file making Dr.Web Security Space functional, click **Get new license**. After that, the registration wizard opens. Note that the registration wizard also opens automatically when Dr.Web Security Space is started for the first time after installation.

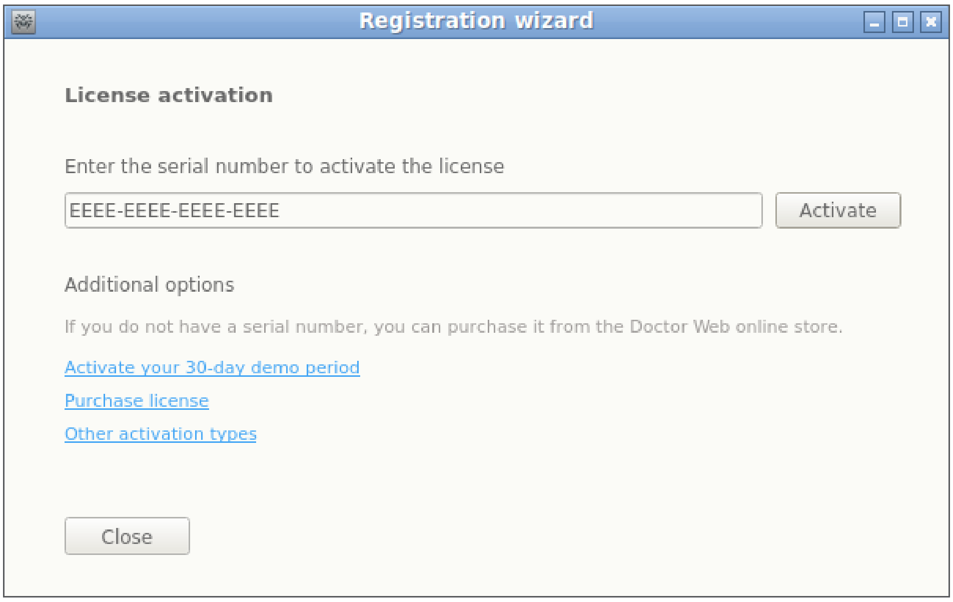
In the first step, you must select an activation type. The following three types are available:

1. Activation of a license or a demo period using an available serial number.
2. Obtaining a demo period.
3. Installation of a key file obtained earlier.

> To register a serial number or to get a demo period, an internet connection is required.

1. Activation of License or Demo Period Using a Serial Number

To activate a license or a demo period using an available serial number, enter it in the text field and click **Activate**.



Registration wizard

License activation

Enter the serial number to activate the license

EEEE-EEEE-EEEE-EEEE

Additional options

If you do not have a serial number, you can purchase it from the Doctor Web online store.

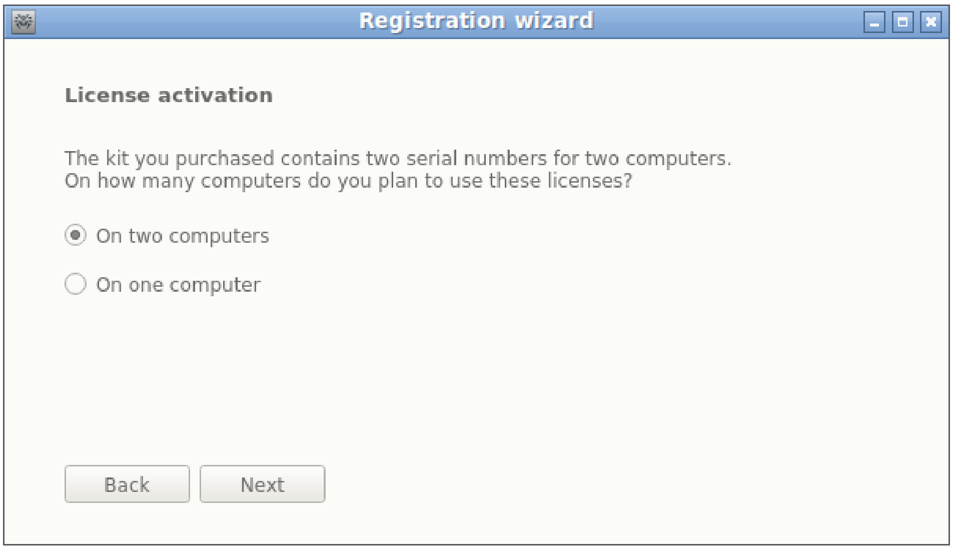
Figure 21. Registration using a serial number

If you do not have a serial number or a valid key file, you can purchase the license at the official website of Doctor Web. To open the online store page, click **Purchase license**.

For information about other ways to purchase the license for Dr.Web products, refer to the Registration and Activation section.

Once you click **Activate**, a connection to the Doctor Web registration server will be established.

If the serial number specified in the first step corresponds to a license for two computers, you need to select the number of computers to run Dr.Web Security Space. If you select **On two computers**, you can activate the second serial number on another computer and receive another license key file. The registered licenses will have the same validity period (for example, one year). If you select **On one computer**, you must specify the second serial number from the kit. In this case, you will not be able to register this serial number later on another computer (or use on this computer a copy of the license key file received after the activation of the combined license), but the duration of the current license will be doubled (for example, extended to two years if the license period is one year).



License activation

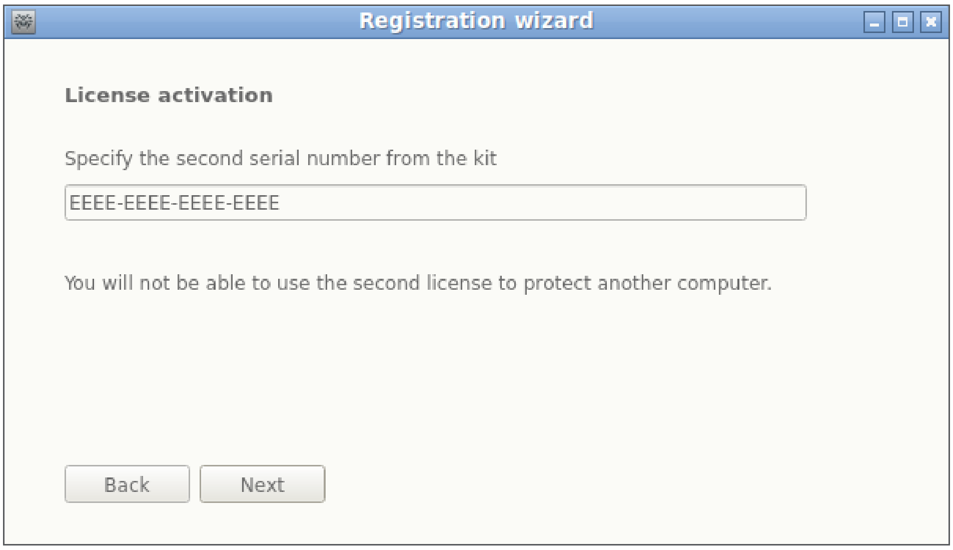
The kit you purchased contains two serial numbers for two computers.
On how many computers do you plan to use these licenses?

● On two computers

○ On one computer

Figure 22. Selecting the number of computers

After selecting the number of computers to activate the license on, click **Next**, and if you have selected **On one computer**, enter the second serial number from the kit in the next step of the wizard, and click **Next** once again.



License activation

Specify the second serial number from the kit

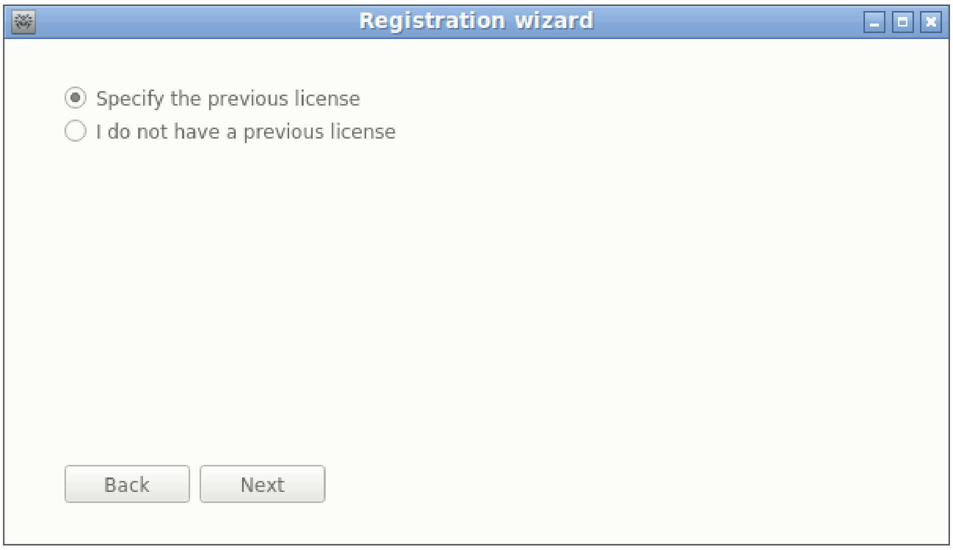
EEEE-EEEE-EEEE-EEEE

You will not be able to use the second license to protect another computer.

Figure 23. Entering the second serial number from the kit

To continue the registration, click **Next**.

In the next step, you are prompted to specify a previous license.



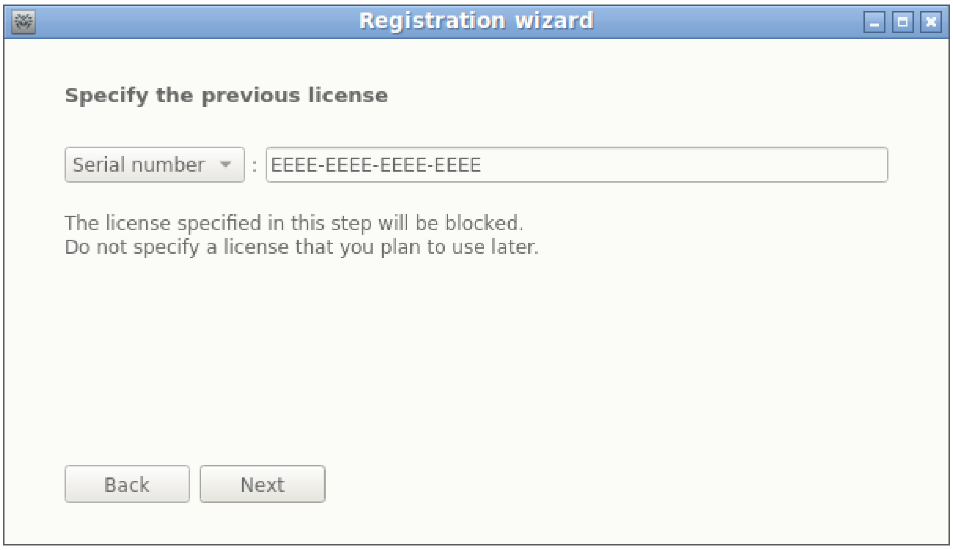
Registration wizard

● Specify the previous license

○ I do not have a previous license

Figure 24. Prompting to specify a previous license

If you have selected **Specify the previous license**, specify the previous license serial number or a path to the corresponding key file in the opened window.



Registration wizard

Specify the previous license

Serial number : EEEE-EEEE-EEEE-EEEE

The license specified in this step will be blocked.
Do not specify a license that you plan to use later.

Figure 25. Specifying a previous license

If you specify a license which is not expired, the new license period will be extended by the remaining period of the previous license.

To specify the previous license, you can either enter its serial number in the corresponding box or specify its key file. The previous license type can be selected in a drop-down list to the left of the input box. To specify the key file, do one of the following:

- enter the path to it in the input box;
- specify the file via a standard file chooser by clicking **Browse**;
- drag and drop the file from the file manager window to the window of the registration wizard.

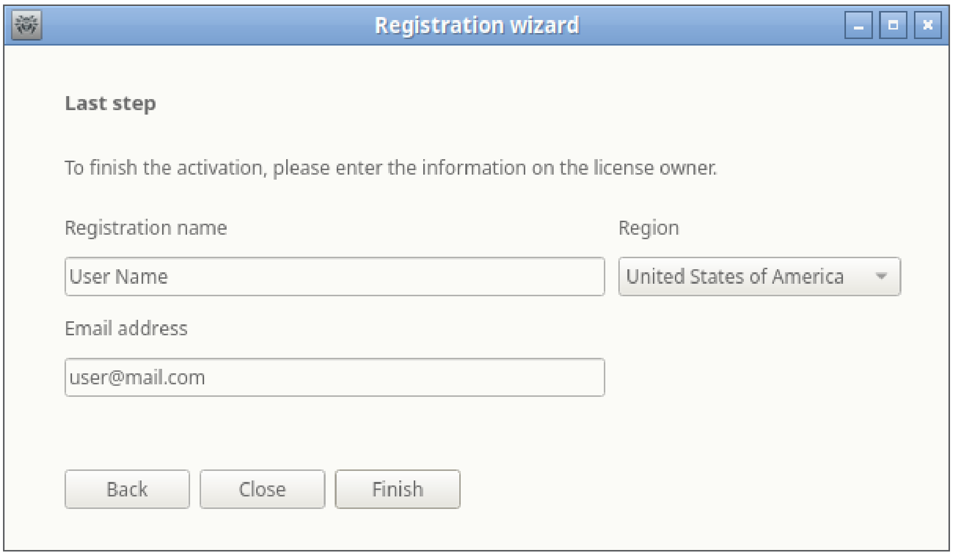
You can specify a `.zip` archive containing the key file without unpacking it.

To continue the registration, click **Next**.

In the next step, specify valid registration information including the following:

- registration name;
- your region (country) selected from the list;
- valid email address.

All registration form fields are mandatory.



Registration wizard

Last step

To finish the activation, please enter the information on the license owner.

Registration name: User Name

Region: United States of America

Email address: user@mail.com

Figure 26. User information page

After all fields are filled in, click **Finish** to connect to the server and obtain the license key file. If necessary, you can use the license key file on another computer after you remove it from the current computer.

2. Obtaining a Demo Period

If you would like to activate a demo period that provides full functionality of Dr.Web Security Space components for a period of 30 days, click the link **Activate your 30-day demo period** in the first step of activation.

When activating the demo period for 1 month via License Manager, you are not required to provide your personal data.

3. Installation of a Key File Obtained Earlier

If you already have a valid license and the corresponding key file (for example, obtained from Doctor Web or its partners via email), you can activate Dr.Web Security Space by installing this key file. For that purpose, click **Other activation types** in the first step of activation and specify the key file path in the displayed input box.

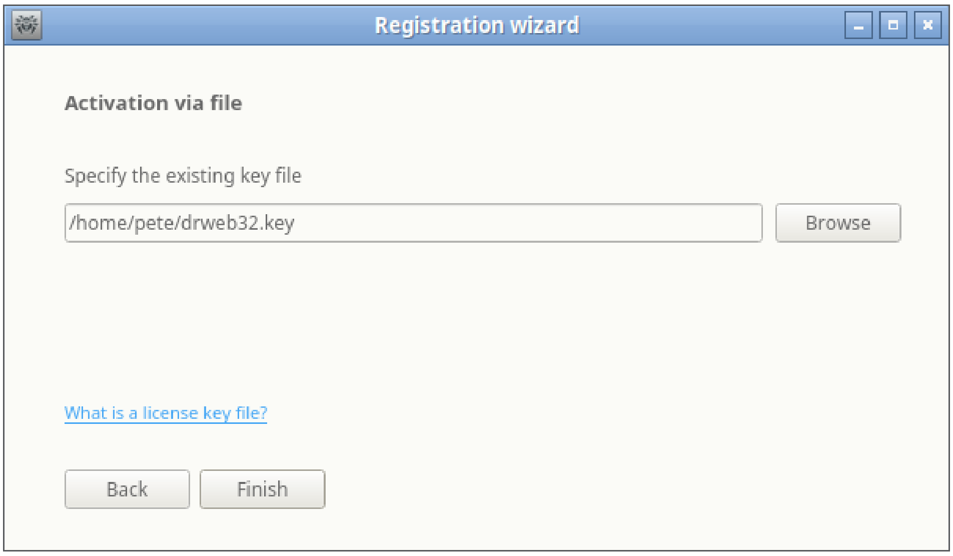


Figure 27. Activation via key file

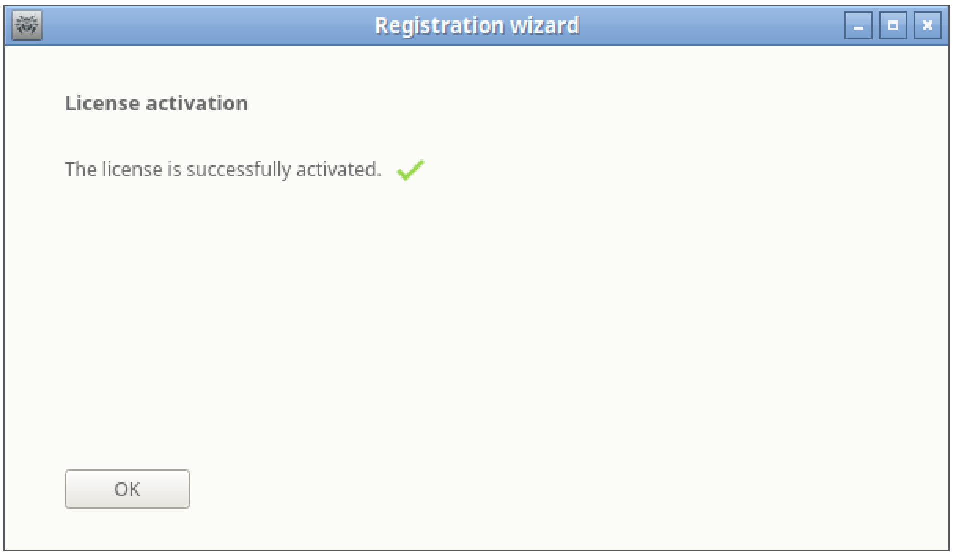
To specify the key file, you can:

- enter the path to it in the input box;
- specify the file via a standard file chooser by clicking **Browse**;
- drag and drop the file from the file manager window to the window of the Registration wizard.

You can specify a `.zip` archive containing the key file without unpacking it.

After you specify the key file path (or the path to the archive containing the key file), click **Finish** to install the key file automatically. If required, the key file is automatically unpacked and copied to the directory with Dr.Web Security Space files. An internet connection is not required in this case.

After the activation procedure completes (using any method described above), the final page of the registration wizard with the notification of successful activation of a license or a demo period is displayed. Click **OK** to exit the registration wizard and return to the main page of Dr.Web Security Space.



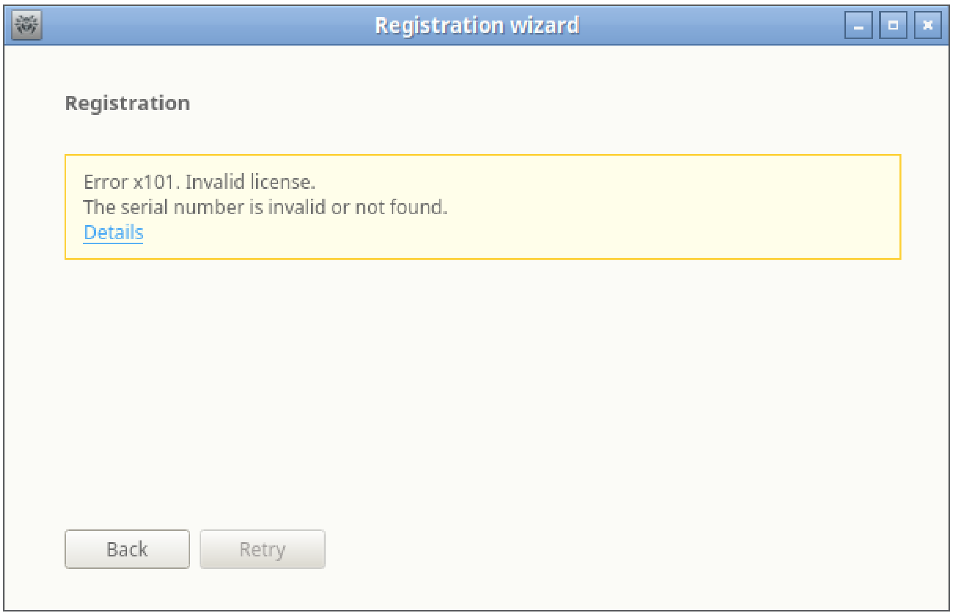
Registration wizard

License activation

The license is successfully activated. ✓

Figure 28. Successful activation notification

If an error occurs in any step of the registration process, a page with the corresponding notification and short error description is displayed. The figure below shows an example of such a page.



Registration wizard

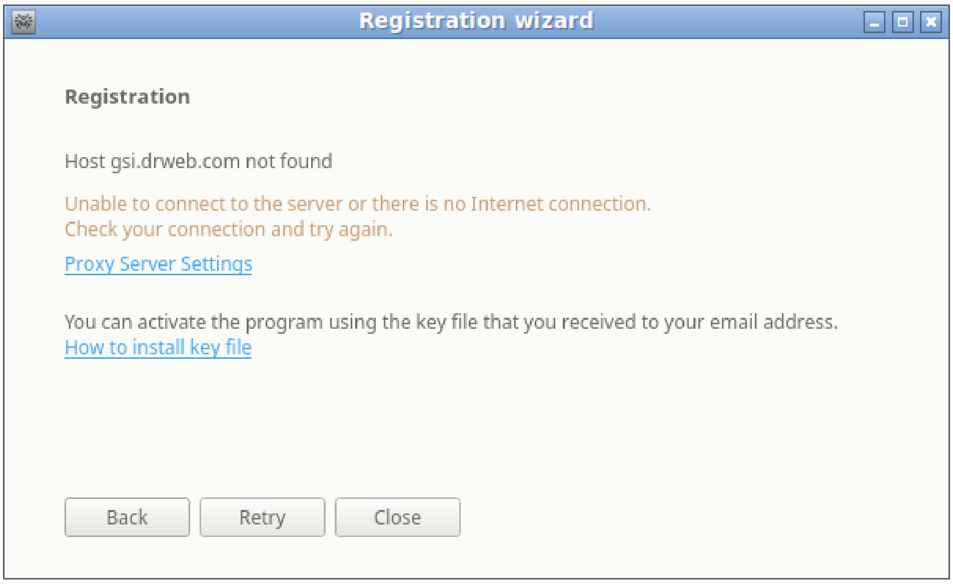
Registration

Error x101. Invalid license.
The serial number is invalid or not found.
Details

Figure 29. Error message

If an error occurs, you can return to the previous step to make corrections (for example, correct the serial number or specify the correct file path). To do this, click **Back**.

If the error is caused by a temporary issue (for example, temporary network failure), you can try to repeat this step by clicking **Retry**. If necessary, click **Close** to cancel the registration and exit the registration wizard. In this case, you will need to retry the registration procedure later. If the wizard cannot establish a connection to the Doctor Web registration server to verify the serial number, the page with the corresponding message is displayed.



Registration

Host gsi.drweb.com not found

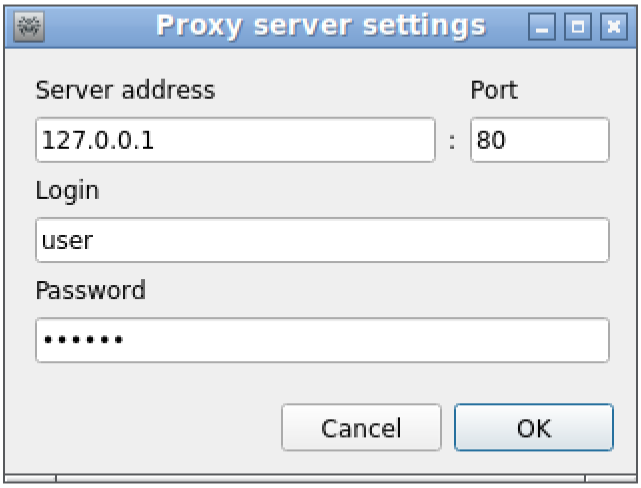
Unable to connect to the server or there is no Internet connection.
Check your connection and try again.

Proxy Server Settings

You can activate the program using the key file that you received to your email address.
How to install key file

Figure 30. Registration server connection error

If the error has occurred because your computer cannot use a direct internet connection, but you use a proxy server to access the internet, click the link **Proxy Server Settings** to open the window containing proxy server settings:



Server address: 127.0.0.1

Port: 80

Login: user

Password: ••••••

Figure 31. Proxy server settings

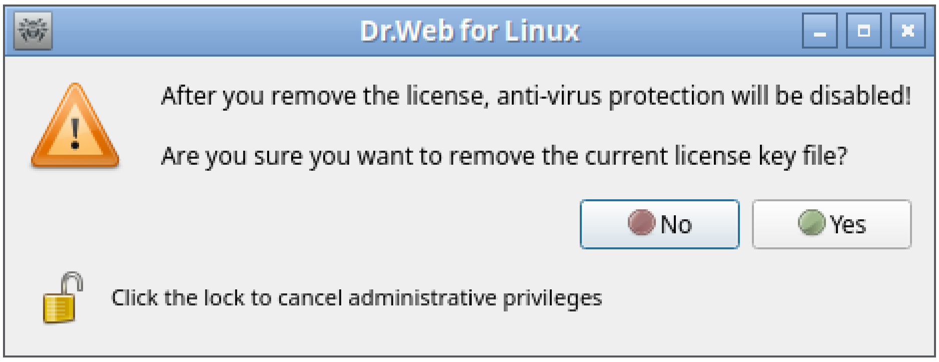
Specify the proxy server settings and click **OK**. After that retry establishing connection to the Doctor Web registration server by clicking **Retry**.

Upon activation of a new license and generation of a new key file, the previous key file used by Dr.Web Security Space is automatically saved as a backup copy in the `/etc/opt/drweb.com` directory. If required, you can use it again by installing the key file.

Deleting License Key File

If necessary (for example, you decided to use Dr.Web Security Space on another computer instead of the current computer), you can delete the installed license key file that manages Dr.Web Security Space operation. For that purpose, open the page with license information (the start page of License manager) and click ✕ next to the number of the current license.

After that, confirm deletion of the license key file from the current computer in the appeared window by clicking **Yes**. If you want to cancel the deletion, click **No**.



After you remove the license, anti-virus protection will be disabled!

Are you sure you want to remove the current license key file?

Click the lock to cancel administrative privileges

Figure 32. Confirming deletion of the key file

To delete the license key file, the application must be started with superuser privileges. If the application does not have elevated permissions, the **Yes** button is unavailable on attempt to delete the key file. If required, you can elevate the application privileges and, if the elevation succeeds, the **Yes** button becomes available.

Deletion of the license key file does not affect the license validity period. If the license is not expired, you can obtain a new key file for this license for the remaining period.

After the license key file is deleted, all anti-virus functions of Dr.Web Security Space (file scanning, updating virus databases, the scanning engine and databases of web resource categories, file system monitoring) are blocked until a new license or a demo period is activated.

8.1.6. Viewing Messages From a Centralized Protection Server

In this section:

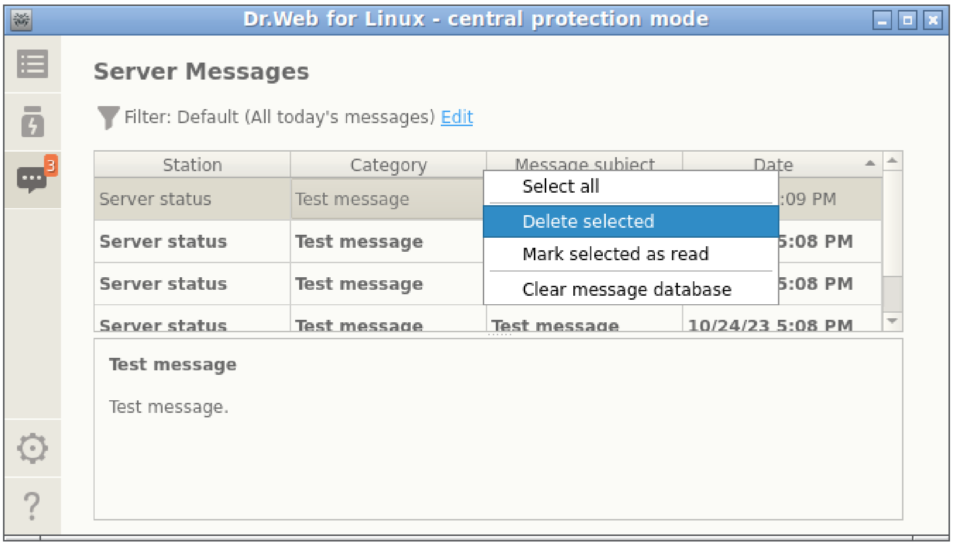
- General Information.
- Applying Actions to Messages.
- Message Filtering.

General Information

If Dr.Web Security Space is connected to a centralized protection server, you can use an interface to view messages on the state of the anti-virus network that are sent by the centralized protection server to the controlled workstations. An anti-virus network administrator can use this tool to monitor the network state and important events related to the operation of the centralized protection server.

> The messages on the network state and events are sent to a workstation only if the anti-virus network administrator configured the corresponding setting on the centralized protection server to which Dr.Web Security Space is connected. Otherwise, the messages cannot be viewed and the corresponding page is not displayed on the Dr.Web Security Space main window.

The interface for viewing messages from the server is displayed on a separate page. To open this page, click 💬 on the navigation pane.



Dr.Web for Linux - central protection mode

Server Messages

Filter: Default (All today's messages) Edit

Station	Category	Message subject	Date
Server status	Test message		:09 PM
Server status	Test message		5:08 PM
Server status	Test message		5:08 PM
Server status	Test message	Test message	10/24/23 5:08 PM

Test message

Test message.

Figure 33. Messages from the centralized protection server

For each message on the list, the following information is available:

- name (address) of the workstation that is mentioned in the message;
- message category;
- message title (subject);
- date and time of sending the message by the server.

To view the message, select it from the list. The text of the selected message will be shown on the pane under the message list. Unread messages are in bold.

The text of the messages about the state and events of the anti-virus network is in the language that is specified in the centralized protection server settings.

Applying Actions to Messages

To apply an action to a message, right-click the row with the information about the message and select the required action in the drop-down list. If you need to apply an action to several messages, select them in the list before activating the drop-down list. To select several rows, hold CTRL or SHIFT:

- When you hold CTRL, messages are selected one by one.
- When you hold SHIFT, messages are selected contiguously.

To select all messages, press CTRL+A.

The drop-down list contains the following actions:

- select all filtered messages in the list;
- delete the selected messages;
- mark the selected messages as read;
- purge the message database.

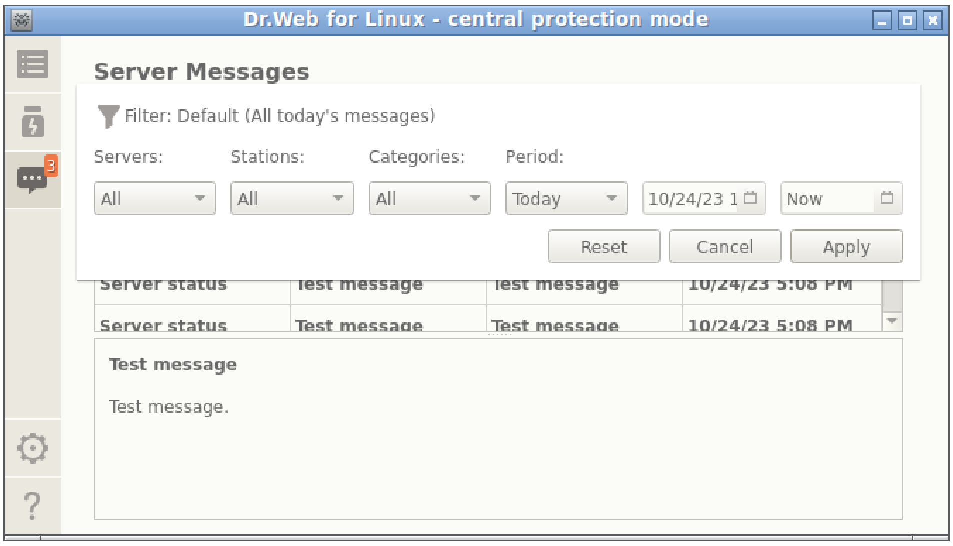
If you purge the database, all received messages are deleted (including unread messages).

The messages received from the centralized protection server are automatically deleted at the end of the maximum storage time that is specified in settings.

Message Filtering

Since the server can send a significant number of messages, you can filter them by a sending server address, an anti-virus network workstation name, a message category or a receiving period. By default, the enabled filter shows all categories of messages received from all servers during the current day.

If necessary, you can edit the message filter. For that, click the **Edit** link. After that, the filter pane opens at the top.



Dr.Web for Linux - central protection mode

Server Messages

Filter: Default (All today's messages)

Servers: All
Stations: All
Categories: All
Period: Today, 10/24/23 1, Now

Reset | Cancel | Apply

Server status	Test message	Test message	10/24/23 5:08 PM
Server status	Test message	Test message	10/24/23 5:08 PM

Test message

Test message.

Figure 34. Message filter pane

On the filter pane, you can specify the following filtering parameters:

- **Servers**—a list of servers from which the messages are shown.
- **Stations**—a list of workstations about which the messages are shown.
- **Categories**—a list of message categories to show.
- **Period**—a generation period of the messages to show. You can select a standard period from the list or you can specify specific start and end time of the generation period.

To save changes to the filter, click **Apply**. To close the filter pane and discard the changes, click **Cancel**. To reset the filter to the default values, click **Reset**.

8.1.7. Managing Application Privileges

Some actions in the Dr.Web Security Space window can be performed only if the application has elevated privileges (*administrative privileges*) that correspond to the *superuser* (the *root* user) privileges. Among such actions are the following:

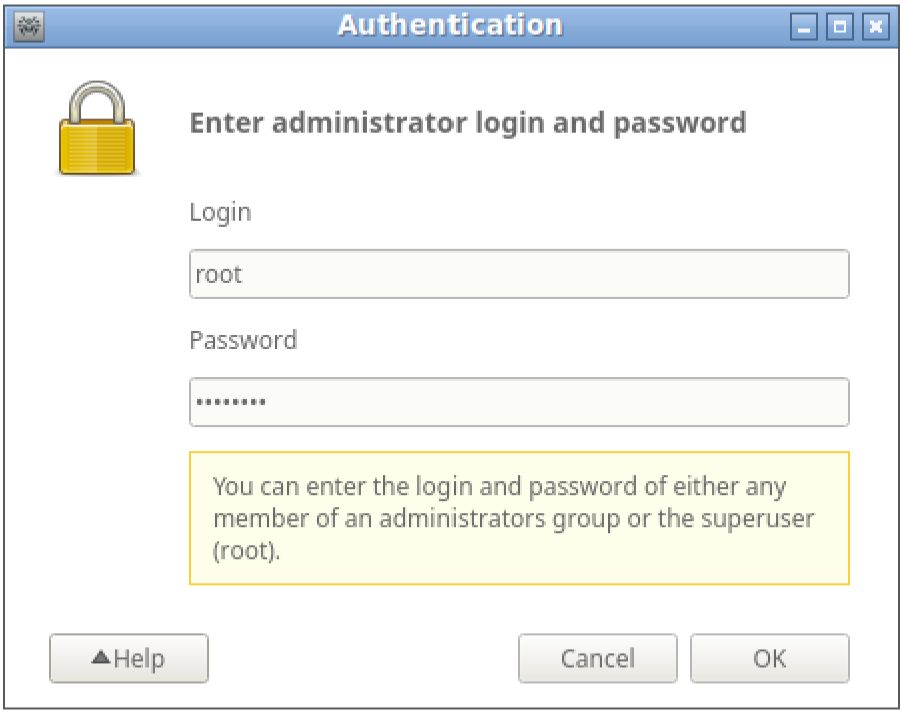
- management of objects put in the system quarantine (that is, in the quarantine directory not owned by the user who started Dr.Web Security Space);
- scanning of files and directories of other users (in particular, the superuser);
- disabling the SpIDer Guard file system monitor;
- disabling the SpIDer Gate network connection monitor;
- removal of a license key file, connection and disconnection from a centralized protection server.

> Even if the application is started by root (for example, by using `su` or `sudo` commands), it *is not* granted elevated privileges by default.

All pages of the Dr.Web Security Space window that provide for actions requiring elevated privileges contain a special button with a lock icon. The icon indicates whether or not the Dr.Web Security Space window currently has superuser privileges:

🔒	The application does not have elevated privileges. Click the icon to elevate the privileges to superuser privileges.
🔓	The application has elevated privileges. Click the icon to lower the privileges; that is, to switch from superuser to user privileges.

Once you click the icon for privilege elevation, the user authentication window opens.



Enter administrator login and password

Login

root

Password

••••••••

You can enter the login and password of either any member of an administrators group or the superuser (root).

Figure 35. Authentication window

To grant superuser privileges to the application, provide the name (login) and password of the user indicated in the Dr.Web Security Space settings as an *administrator group*, or the login and password of the superuser (the *root* user) and click **OK**. To cancel the privilege elevation, close the window by clicking **Cancel**. To view or hide a hint about authentication, click **Help**.

By default, during the installation of Dr.Web Security Space, a system group of users who can elevate their rights to superuser privileges (for example, the *sudo* group) is selected as the administrator group. If the name of such system group cannot be determined, you can enter the superuser (the *root* user) login and password in the authentication window to elevate application privileges.

Switching from administrative privileges to user rights does not require authentication.

8.1.8. Help and Reference

To access the Help file, click ? on the navigation pane of Dr.Web Security Space.

The following drop-down menu will appear:

- **Help**—open the Dr.Web Security Space User manual.
- **Forum**—open the webpage of the Doctor Web forum (requires an internet connection).
- **Technical support**—open the Doctor Web technical support webpage (requires an internet connection).
- **My Dr.Web**—open your personal webpage on the Doctor Web official website (requires an internet connection).
- **About**—open a window with information about your version of Dr.Web Security Space.

Additionally, when a page of the Dr.Web Security Space main window displays an error message, you can follow the **Details** link to get information about the error and instructions to resolve the issue.

8.1.9. Operation Settings

You can configure the following operation parameters in Dr.Web Security Space settings window:

- update frequency;
- reaction of Dr.Web Security Space to threats detected during scanning on demand by Scanner or detected by the SpIDer Guard file system monitor;
- a list of objects excluded by Scanner and SpIDer Guard from scanning;
- parameters of monitoring of network connections;
- schedule of scans performed by Scanner;
- protection mode (standalone or centralized);
- using the Dr.Web Cloud service.

To open the settings window, click ⚙ on the navigation bar.

The following pages are available in the settings window:

- Main allows to configure notifications and frequency of automatic updates.
- Scanner allows to configure reaction of Dr.Web Security Space to threats detected by Scanner during scans on demand or scheduled scans.
- SpIDer Guard allows to configure reaction of Dr.Web Security Space to threats detected by the SpIDer Guard file system monitor.

- SpIDer Gate allows to configure how the SpIDer Gate monitor controls network connections.
- Exclusions allows to configure the list of objects to be excluded from scans on demand or scheduled scans, as well as from the list of objects monitored by SpIDer Guard and controlled by SpIDer Gate.
- Scheduler allows to configure periodical scanning according to the specified schedule.
- Network allows to enable or disable a mode of scanning secure network connections (based on SSL/TLS, such as HTTPS) for SpIDer Gate, to save the Dr.Web certificate, which is used to intercept secure network connections, to a file.
- Mode allows to select a protection mode (standalone or centralized protection) in which Dr.Web Security Space operates.
- Dr.Web Cloud allows or prohibits Dr.Web Security Space to use the Dr.Web Cloud service.

To open the help file, click ? on the corresponding page of the settings window.

All settings changed on these pages are applied immediately.

If Dr.Web Security Space operates in centralized protection mode, some settings can be blocked and unavailable for modification.

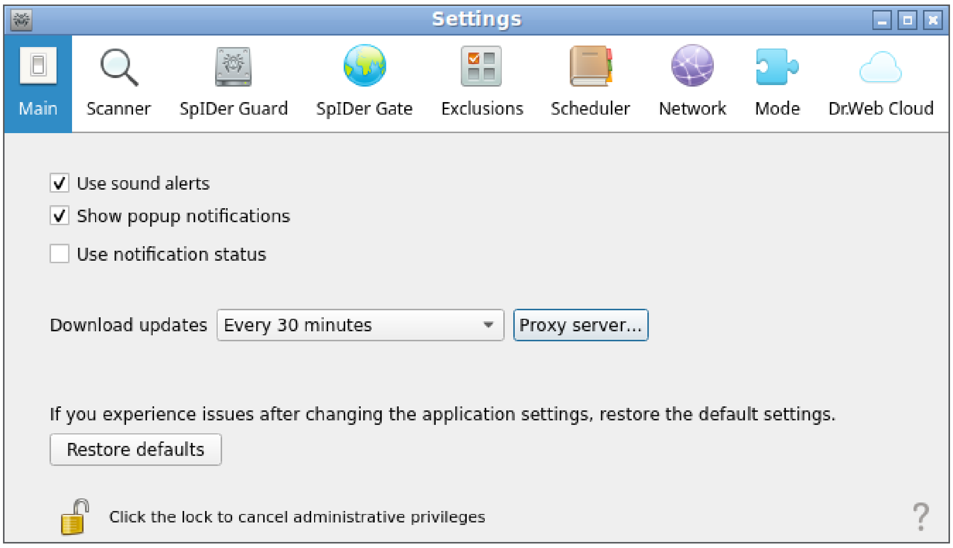
8.1.9.1. Main Settings

In this section:

- General Information.
- Configuring Proxy Server for Updates.

General Information

On the **Main** tab, you can configure the main application settings.



☑ Use sound alerts

☑ Show popup notifications

□ Use notification status

Download updates: Every 30 minutes

If you experience issues after changing the application settings, restore the default settings.

Click the lock to cancel administrative privileges

Figure 36. Main settings

Control	Action
Check box **Use sound alerts**	Select this check box if you want Dr.Web Security Space to use sound notifications on such events as: • threat detected (by Scanner or SpIDer Guard); • object scan error; • and so on.
Check box **Show popup notifications**	Select this check box if you want Dr.Web Security Space operating in graphical mode to show pop-up notifications on particular events, such as • date/time of detecting the threat; • scan error; • and so on.
Check box **Use notification status**	Select this check box if you want Dr.Web Security Space to show pop-up notifications when the status of the components changes (for example, they are enabled or disabled). An alternative mechanism of showing pop-up notifications is used, which may be useful, for example, if the desktop environment does not support showing the icon of Dr.Web Security Space in the notification area. If this feature is not supported by the desktop environment, this check box will be absent on the **Main** tab.
Drop-down list **Download updates**	Select the frequency with which virus databases, databases of web resource categories and the scanning engine are automatically updated.

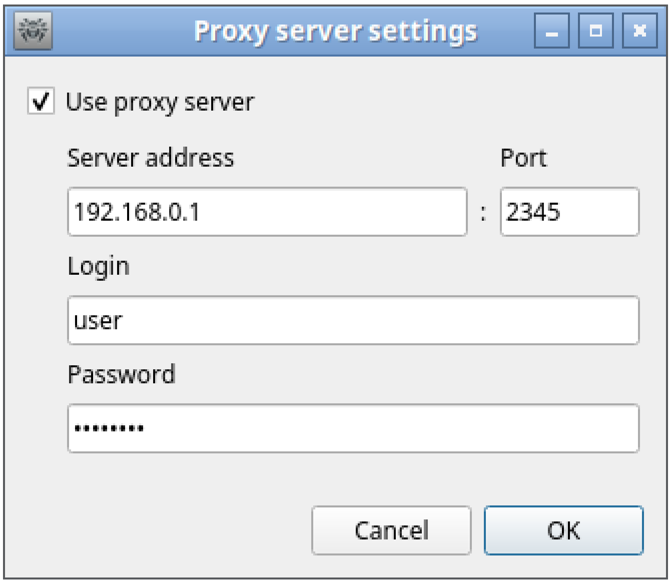
Control	Action
Button **Proxy server...**	Open a window to configure a proxy server for receiving updates. The proxy server may be needed if connecting to external servers is prohibited by network security policies.
Button **Restore defaults**	Click to restore default settings.

To manage update settings and restore defaults, the application must have root privileges (refer to the Managing Application Privileges section).

Configuring Proxy Server for Updates

In the window to configure a proxy server for receiving updates, you can adjust the following parameters:

- enable or disable the use of the proxy server for receiving updates;
- an address of the proxy server to be used for receiving updates;
- a port for connecting to the proxy server;
- a login and a password used for authentication on the proxy server.



☑ Use proxy server

Server address: 192.168.0.1

Port: 2345

Login: user

Password: ••••••••

Figure 37. Proxy server settings

The proxy-server address and port are mandatory parameters. As the address, you can specify either an IP address or FQDN of the host on which the proxy server operates. Since updates are received via HTTP, an HTTP proxy server must be used. You must specify a login and a password only if the HTTP proxy server requires authorization.

To save the changes and close the window, click **OK**. To discard the changes and close the window, click **Cancel**.

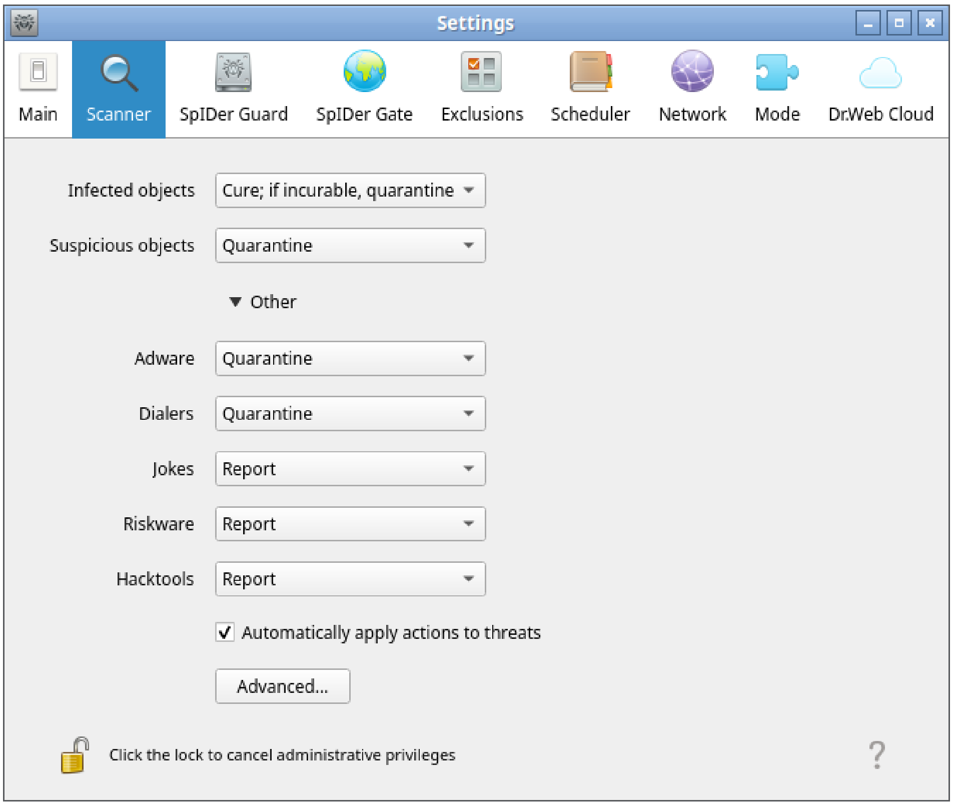
8.1.9.2. File Scanning Settings

In this section:

- General Information.
- Advanced File Scanning Settings.

General Information

On the **Scanner** tab, you can configure actions to be applied by Dr.Web Security Space to threats detected by Scanner while scanning files on demand or on schedule.



Infected objects: Cure; if incurable, quarantine

Suspicious objects: Quarantine

▼ Other

Adware: Quarantine

Dialers: Quarantine

Jokes: Report

Riskware: Report

Hacktools: Report

☑ Automatically apply actions to threats

Click the lock to cancel administrative privileges

Figure 38. Scanner settings

Select actions from the drop-down lists to be applied by Dr.Web Security Space to objects upon detection of any threat of the corresponding type.

> If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined and not deleted.

By selecting the check box **Automatically apply actions to threats**, you instruct Dr.Web Security Space to apply the specified action to a threat once it is detected by Scanner while scanning on demand or on schedule (you will be informed about threat neutralization, and threat details will be available on the threat list). If the check box is cleared, the threat detected by Scanner will be added to the list of detected threats and you will need to manually select the action to be applied to the object containing the threat.

Click **Advanced** to open the window with advanced file scanning settings.

Notes:

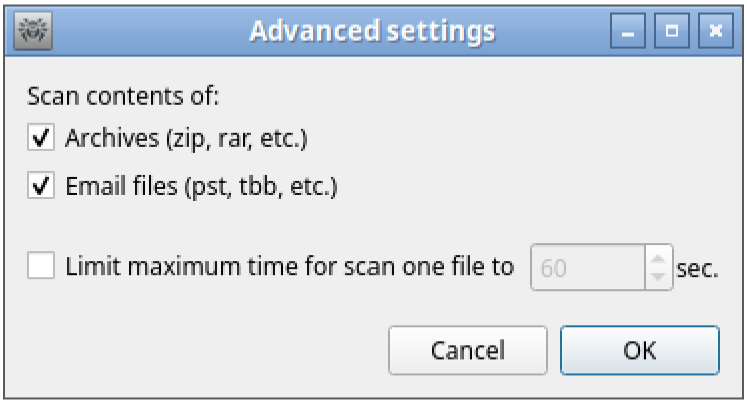
- You can exclude files and directories from scanning by Scanner on the **Exclusions** tab.
- Reactions to threat detection defined for Scanner, including automatically applying actions, do not have an effect on the behavior of the SpIDer Guard monitor. Its reactions are specified on the corresponding page.

> To change the reaction of Scanner to threats and access advanced settings, the application must operate with elevated permissions (refer to the Managing Application Privileges section).
>
> A possibility to configure Scanner when Dr.Web Security Space is controlled by a centralized protection server can be blocked if this is not allowed by the server.

Advanced File Scanning Settings

You can configure the following parameters of Scanner on the window with advanced scanning settings:

- Enable and disable the scanning of containers:
  - archives;
  - mail files.
- Set a time limit for scanning one file.



Scan contents of:

☑ Archives (zip, rar, etc.)

☑ Email files (pst, tbb, etc.)

☐ Limit maximum time for scan one file to 60 sec.

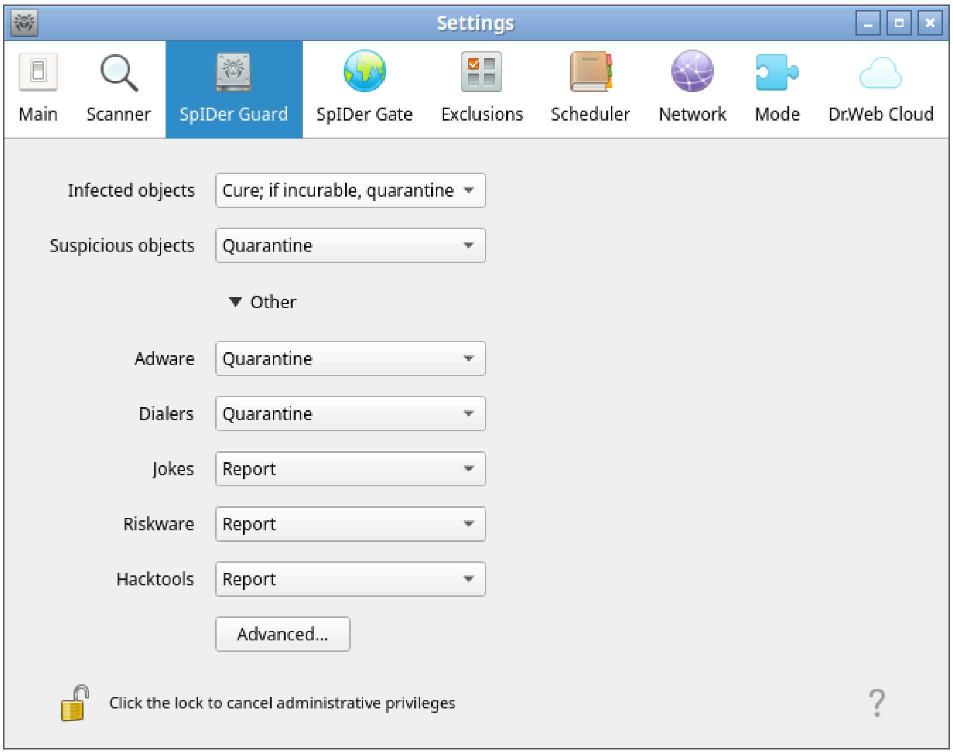
Figure 39. Advanced file scanning settings

If the check boxes that enable the scanning of containers are not selected, the container files are scanned by Scanner anyway, but enclosed files are excluded from scanning.

To save the changes and close the window, click **OK**. To discard the changes and close the window, click **Cancel**.

8.1.9.3. File System Monitoring Settings

On the **SpIDer Guard** tab, you can configure actions to be applied by Dr.Web Security Space to threats detected by the SpIDer Guard file system monitor.



Infected objects: Cure; if incurable, quarantine

Suspicious objects: Quarantine

▼ Other

Adware: Quarantine

Dialers: Quarantine

Jokes: Report

Riskware: Report

Hacktools: Report

Figure 40. File system monitoring settings

This tab, including the window with advanced settings, is similar to the **Scanner** tab where file scanning settings can be adjusted.

If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined and not deleted.

Notes:

- You can exclude files and directories from SpIDer Guard monitoring on the **Exclusions** tab.
- For details on enabling the enhanced file monitoring mode for SpIDer Guard, refer to the File Monitoring Modes section.
- Reactions to threat detection specified for the SpIDer Guard monitor do not have an effect on the behavior of Scanner. The Scanner reactions are specified on the corresponding tab.

To change the settings of the SpIDer Guard file system monitor, the application must operate with elevated permissions (refer to the Managing Application Privileges section).

A possibility to configure SpIDer Guard when Dr.Web Security Space is controlled by a centralized protection server can be blocked if this is not allowed by the server.

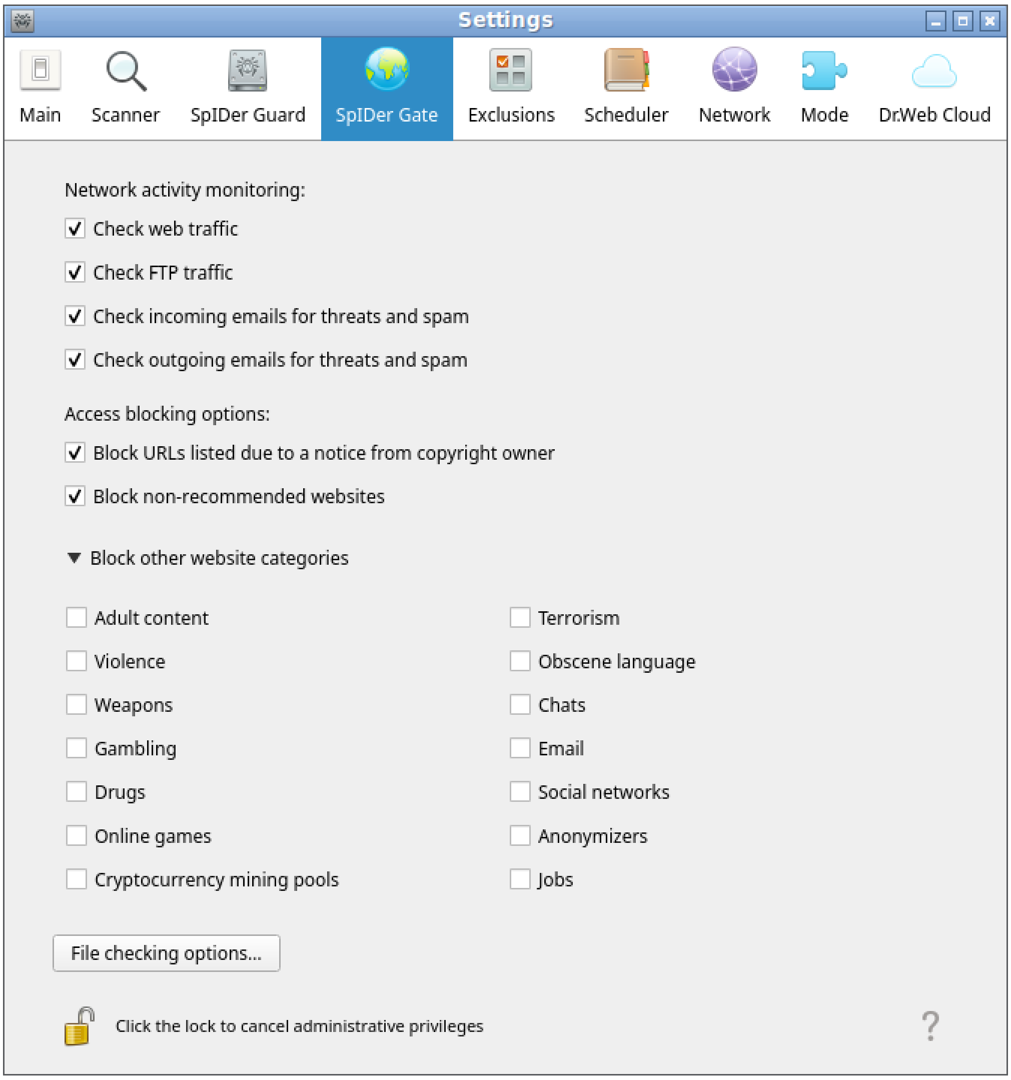
8.1.9.4. Network Connection Monitoring Settings

In this section:

- General Information.
- Website Category Selection.
- Managing File Scanning Parameters.

General Information

On the **SpIDer Gate** page, you can configure security policies used by the SpIDer Gate network connection monitor upon an attempt to access the internet.



Network activity monitoring:

☑ Check web traffic

☑ Check FTP traffic

☑ Check incoming emails for threats and spam

☑ Check outgoing emails for threats and spam

Access blocking options:

☑ Block URLs listed due to a notice from copyright owner

☑ Block non-recommended websites

▼ Block other website categories

□ Adult content

□ Violence

□ Weapons

□ Gambling

□ Drugs

□ Online games

□ Cryptocurrency mining pools

□ Terrorism

□ Obscene language

□ Chats

□ Email

□ Social networks

□ Anonymizers

□ Jobs

Figure 41. Internet access control settings

By selecting or clearing check boxes in the **Network activity monitoring** section, you can define types of network activity to be controlled by the monitor, if it is enabled.

Website Category Selection

Check boxes in the **Monitoring options** section define categories of websites and hosts to be blocked (this applies not only to attempts to access such websites using a browser, but also to attempts to access FTP servers). By selecting or clearing corresponding check boxes, you can allow or block access to websites and hosts from the following categories:

Category	Description
URLs added due to a notice from copyright owner	Websites with content that infringes copyright (according to the copyright holder of this content). Among those are pirated websites, file link catalogs, file hosting services and others.

Category	Description
Non-recommended websites	Websites with unreliable content (suspected of phishing, password theft and so on)
Adult content	Websites with adult content
Violence	Websites that contain violent material (for example, war scenes, acts of terrorism and so on)
Weapons	Websites that contain information about weapons and explosives
Gambling	Internet casinos, gambling and bookmaking websites
Drugs	Websites that contain information about drug production or use
Obscene language	Websites with obscene language
Chats	Chat websites
Terrorism	Websites that contain information about terrorism
Email	Websites that offer free email registration
Social networks	Social networking websites
Online games	Websites that provide access to games using a permanent internet connection
Anonymizers	Websites that allow the user to hide personal information and that provide access to blocked web resources
Cryptocurrency mining pools	Websites that provide access to common services for cryptocurrency mining
Jobs	Job search websites

A database of web resource categories is provided with Dr.Web Security Space and updated automatically together with virus databases. Users do not have permissions to edit the databases of web resource categories.

The same website can belong to several categories. The SpIDer Gate network connection monitor blocks access to a website or a host if it belongs to at least one of the blocked categories. Click the **Block other website categories** label to expand or collapse the list of available categories.

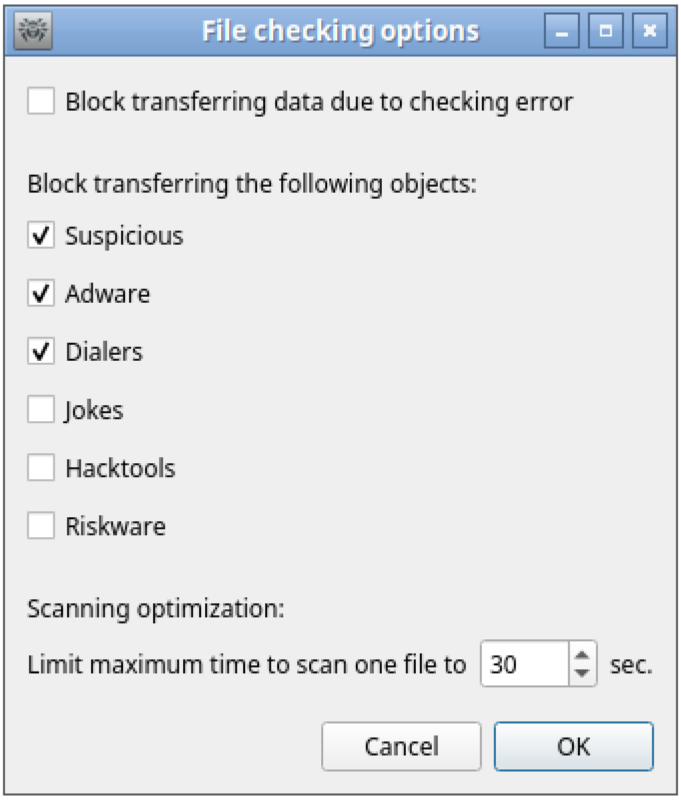
If you need to block access to a website or a host which does not belong to any of these categories, add it to the black list. If, otherwise, you need to allow access to a website or a host which belongs to an unwanted category, add such resource to the white list. You can also configure a list of applications which network connections will not be monitored by SpIDer Gate.

You can configure black and white lists of websites and applications to be excluded by the SpIDer Gate monitor from monitoring on the **Exclusions** page.

As for a special category, *Websites known as infection sources*, access to websites and hosts belonging to this category is always blocked even if they are added to the white list.

Managing File Scanning Parameters

To configure the parameters used by the SpIDer Gate monitor to scan files downloaded from the internet, click **File checking options**.



File checking options

□ Block transferring data due to checking error

Block transferring the following objects:

☑ Suspicious

☑ Adware

☑ Dialers

□ Jokes

□ Hacktools

□ Riskware

Scanning optimization:

Limit maximum time to scan one file to 30 sec.

Figure 42. File scanning settings

In the appeared window, you can specify categories of malicious objects to be blocked on attempt to transfer them. If the check box is selected, attempts to download files that contain a threat of the corresponding category are blocked. If the check box is cleared, files that belong to this category can be downloaded from the internet. You can also set a time limit for scanning downloaded files. If the **Block transferring data due to checking error** check box is selected, attempts to download files that were not scanned owing to an error are blocked. To allow downloading of files that were not scanned, clear this check box (not recommended).

If a downloaded file cannot be scanned because the time limit for scanning it has been reached, such file *will not* be treated as not scanned and will not be blocked even if the **Block transferring data due to checking error** check box is selected.

To save the changes and close the window, click **OK**. To discard the changes and close the window, click **Cancel**.

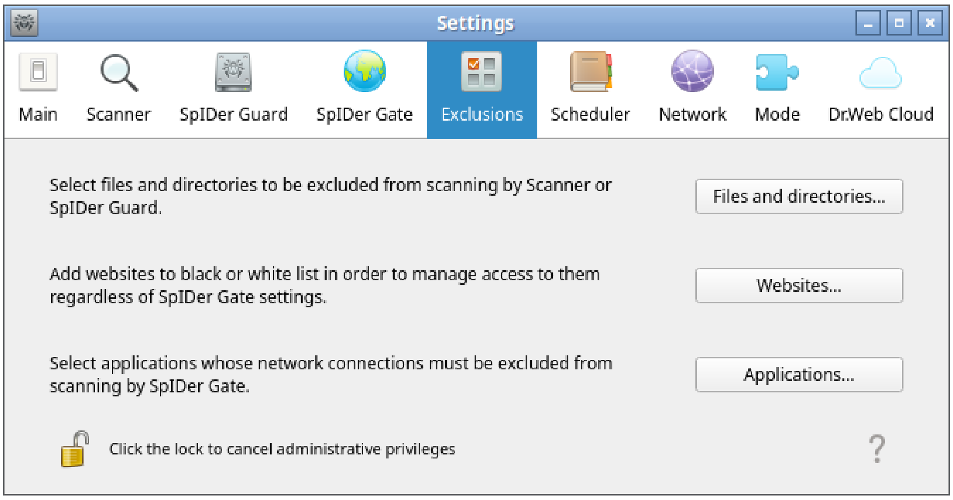
To change the SpIDer Gate network connection monitor settings, the application must be run with elevated privileges, refer to Managing Application Privileges.

Network connection monitoring rules can be edited manually. To configure the component manually, refer to the Administrator Manual for Dr.Web for UNIX Internet Gateways, the section Dr.Web for UNIX Internet Gateways Components ⇒ Dr.Web Firewall for Linux ⇒ Configuration Parameters ⇒ Rules for Traffic Monitoring and Blocking of Access. Note that manual editing can affect the standard filter configuration performed earlier using the graphical interface.

8.1.9.5. Configuring Exclusions

On the **Exclusions** tab, you can see the following buttons for configuring exclusions:

- **Files and directories**—open the window with a list of paths to file system objects that are excluded from scanning by Scanner and the SpIDer Guard file system monitor.
- **Websites**—open the window to manage black and white lists of websites, access to which is regulated regardless of blocking policies set for the SpIDer Gate network connection monitor.
- **Applications**—open the window with a list of applications which network connections will not be controlled by the SpIDer Gate network connection monitor.



Select files and directories to be excluded from scanning by Scanner or SpIDer Guard.

Add websites to black or white list in order to manage access to them regardless of SpIDer Gate settings.

Select applications whose network connections must be excluded from scanning by SpIDer Gate.

Click the lock to cancel administrative privileges

Figure 43. Configuring exclusions

To add or remove objects from the exclusion list, the application must operate with elevated permissions (refer to the Managing Application Privileges section).

8.1.9.5.1. Excluding Files and Directories

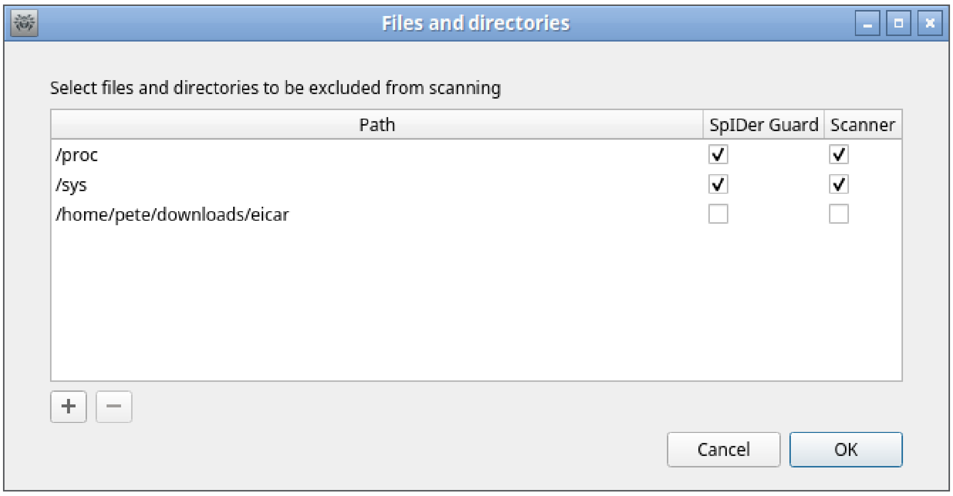
In this section:

- General Information.
- Adding and Removing Objects From the List of Exclusions.

General Information

You can manage the list of files and directories to be excluded from scanning in the **Files and directories** window. To open it, click **Files and directories** on the **Exclusions** tab.

Here you can list paths to objects that you want to exclude from scanning by Scanner at user request and/or as scheduled and from monitoring performed by the SpIDer Guard file system monitor. If a directory is specified, all directory contents is skipped, including subdirectories and nested files.



Select files and directories to be excluded from scanning

Path	SpIDer Guard	Scanner
/proc	☑	☑
/sys	☑	☑
/home/pete/downloads/eicar	□	□

Figure 44. Configuring file and directory exclusions

The same object can be excluded from scanning by Scanner (at request or as scheduled) and from monitoring by the SpIDer Guard file system monitor. The check box in the corresponding column indicates what group of exclusions the object is added to.

Adding and Removing Objects From the List of Exclusions

- To add an object on the list to the group of exclusions for Scanner or for SpIDer Guard, select the corresponding check box in the row of the object. To remove an from the list of exclusions for Scanner or for SpIDer Guard, clear the corresponding check box.

- To add a new object to the list, click + below the list and select the required object in the appeared dialog for selecting directories and files. You can also add objects to the list by dragging them from the file manager window.
- To remove an object from the list, select the corresponding line in the list and click – below the list.

To save the changes and close the window, click **OK**. To discard the changes and close the window, click **Cancel**.

8.1.9.5.2. Exclusion of Applications

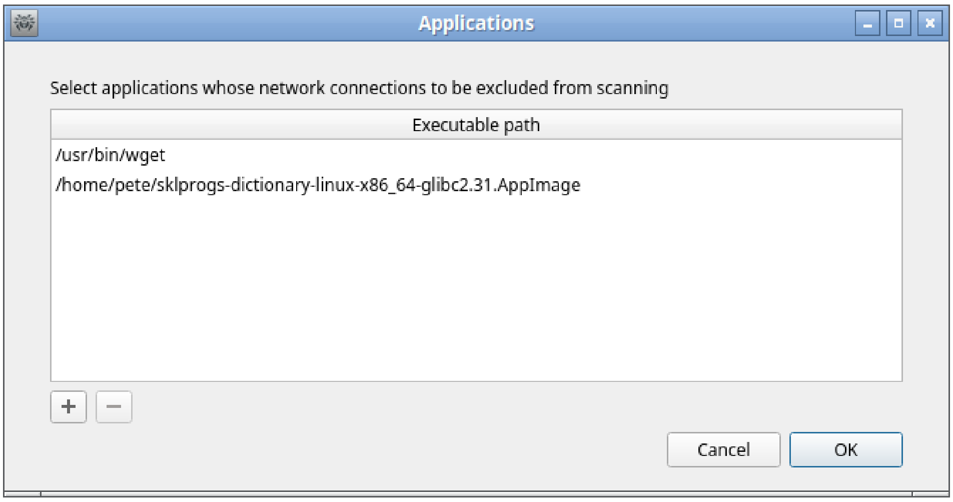
In this section:

- General Information.
- Adding and Removing Applications From the List of Exclusions.

General Information

You can exclude application network connections from monitoring by SpIDer Gate network connection monitor. To do it, open the **Applications** window by clicking the **Applications** button located on the **Exclusions** tab.

Here you can list paths to the application executable files, which network connections should not be controlled by SpIDer Gate network connection monitor.



Select applications whose network connections to be excluded from scanning

Executable path
/usr/bin/wget
/home/pete/sklprogs-dictionary-linux-x86_64-glibc2.31.AppImage

Figure 45. Configuring exclusions for network applications

Adding and Removing Applications From the List of Exclusions

- To add a new application to the list, click + below the list and select the application executable file in the appeared window. In addition, you can add applications to this list by dragging the executable files from the file manager window.
- To remove the application from the list, select the corresponding line in the text and click – below the list.

To close the window and save the changes, click **OK**. To discard the changes and close the window, click **Cancel**.

8.1.9.5.3. Black and white Lists of Websites

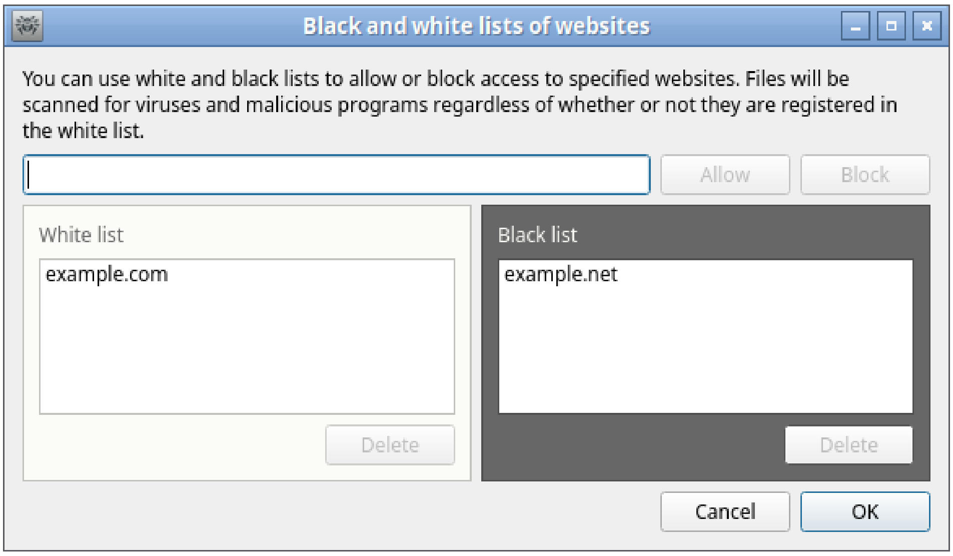
In this section:

- General Information.
- Adding and Removing Websites From the Black and White Lists.

General Information

You can manage black and white lists of websites in the **List Management** window. To open it, click **Websites** on the **Exclusions** tab.

Here you can list the websites, access to which will be always disabled or, on the contrary, always enabled by the SpIDer Gate network connection monitor.



Black and white lists of websites

You can use white and black lists to allow or block access to specified websites. Files will be scanned for viruses and malicious programs regardless of whether or not they are registered in the white list.

Figure 46. Black and white list management window

As for a special website category *Websites known as infection sources*, access to these websites is always disabled even if they are added to the white list.

Adding and Removing Websites From the Black and White Lists

- To add a website to the black or to the white list, type its domain in the edit box and click the respective button.
  - By clicking the **Allow** button, you add the required address to the *white* list.
  - By clicking the **Block** button, you add the required address to the *black* list.
- Adding a domain address to the white or to the black list allows or, otherwise, denies access to all resources within the domain.

- To remove the website from white or black list, select it on the list and click **Delete**.

To close the window and save the changes, click **OK**. To discard the changes and close the window, click **Cancel**.

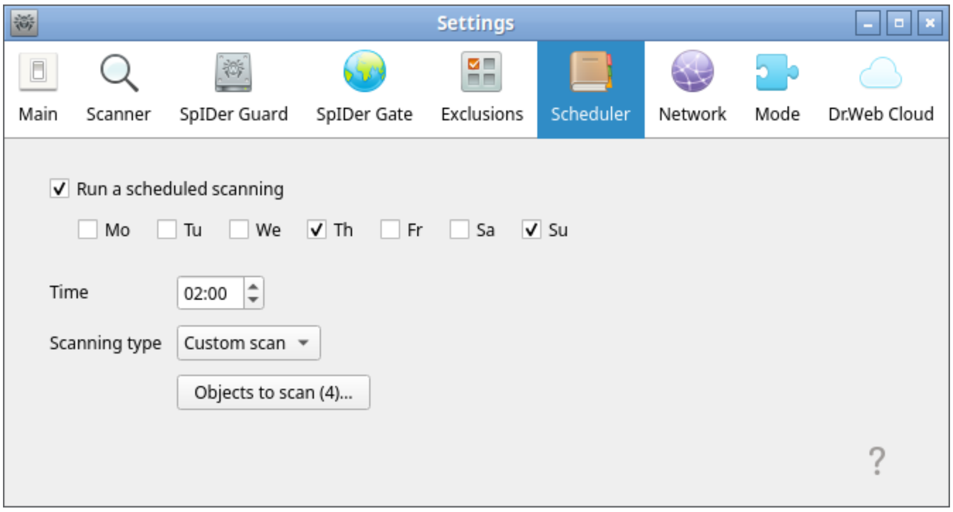
8.1.9.6. Scheduler Settings

In this section:

- General Information.
- Scheduler Settings.

General Information

On the **Scheduler** tab, you can enable starting scheduled scan tasks automatically as well as adjust the schedule and select a scan type.



☑ Run a scheduled scanning

□ Mo □ Tu □ We ☑ Th □ Fr □ Sa ☑ Su

Time 02:00

Scanning type Custom scan

Figure 47. Scheduler settings

To enable automatic scheduled scans, select the check box **Run a scheduled scanning**. In this case, Dr.Web Security Space generates a schedule to periodically start a scan of a certain type.

> Scheduled scanning is started at the specified intervals by the notification agent or directly by the graphical management interface if it is started at the same time with scanning. Scheduled scanning is not started if Dr.Web Security Space is controlled by a centralized protection server or an active license is unavailable.
>
> Scheduled scanning as well as scanning on demand is configured on the **Scanner** tab.

Scheduler Settings

If scheduled scanning is enabled, you can configure the following parameters:

- select the days of the week to start scanning (to do this, select the corresponding check boxes);
- set the time (hours and minutes) to start scanning;
- select a scan type (*Express scan, Full scan* or *Custom scan*).

  If you have selected *Custom scan*, you should also specify a list of objects to be scanned. To do that, click **Objects to scan** (a number of objects to be scanned is indicated in brackets). After that, a custom scan window appears which is similar to a window of custom scan on demand. You can add objects to the list either by clicking + or by dragging and dropping them from the file manager window.

To disable automatic scheduled scanning, clear the check box **Run a scheduled scanning**. The respective task for the notification agent will be automatically removed.

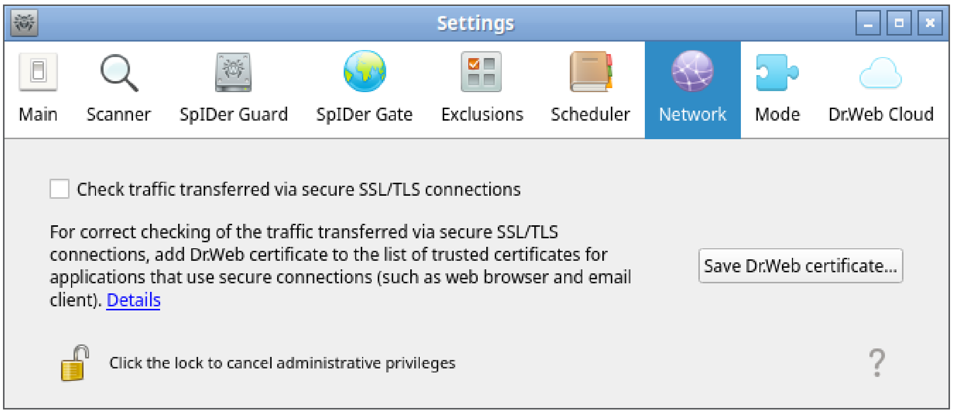
8.1.9.7. Protection Against Threats Distributed over Network

In this section

- General Information
- Configuring Scanning of Secure Network Connections
- Adding Dr.Web Certificate to Lists of Trusted Certificates for Applications
- Adding Dr.Web Certificate to List of Trusted Certificates via Command Line

General Information

On the **Network** tab, you can enable the SpIDer Gate network connection monitor to scan traffic transmitted via secure connections that use SSL- and TLS-based protocols.



□ Check traffic transferred via secure SSL/TLS connections

For correct checking of the traffic transferred via secure SSL/TLS connections, add Dr.Web certificate to the list of trusted certificates for applications that use secure connections (such as web browser and email client). Details

Click the lock to cancel administrative privileges

Figure 48. Protection against threats distributed over the network

Configuring Scanning of Secure Network Connections

To allow SpIDer Gate to scan traffic transmitted via secure network connections that use SSL- and TLS-based protocols, select the check box **Check traffic transferred via secure SSL/TLS connections**. To disable the scanning of secure traffic, clear the check box.

> To manage the scanning of secure traffic, the application must operate with elevated permissions (refer to Managing Application Privileges).
>
> If a mail client is running (such as Mozilla Thunderbird), restart it after the mode **Check traffic transferred via secure SSL/TLS connections** is enabled.

To ensure correct scanning of the traffic transmitted via secure network connections, export the custom Dr.Web certificate to a file and then manually add it to the lists of trusted certificates for applications that use secure connections. Such applications are primarily web browsers and mail clients. Otherwise, if the Dr.Web certificate is not added to the list of trusted certificates, data will be displayed incorrectly if received from a website accessible via HTTPS (for example, from online banking websites, web interfaces of mail servers). If the Dr.Web certificate is not added to the list of trusted certificates for the mail client, authorization on mail servers that use secure protocols (such as SMTPS) for email transmission will fail.

To export the Dr.Web certificate to a file, click **Save Dr.Web certificate**; specify a path to save the file in the appeared window. By default, the file name is `SpIDer Gate Trusted Root Certificate.pem`, but you can change it if necessary.

Then manually add the saved file of the Dr.Web certificate to the lists of trusted certificates for those applications that fail when trying to establish secure connections. You need to add the certificate only once for an application. If you clear and then select the check box **Check traffic transferred via secure SSL/TLS connections** again on the **Network** setting page, you will not need to save the Dr.Web certificate once again or add it to the list of trusted certificates.

Adding Dr.Web Certificate to Lists of Trusted Certificates for Applications

Mozilla Firefox browser

1) Select **Preferences** on the main menu and then (on the appeared settings page) select **Advanced**. Another page opens, where you need to select **Certificates**.
2) Click **View Certificates**. In the appeared window, open the **Authorities** tab and click **Import**.
3) In the appeared window, specify a path to the Dr.Web certificate (by default, its file name is `SpIDer Gate Trusted Root Certificate.pem`) and click **Open**.

4) In the appeared window, use the check boxes to specify a necessary trust level for the certificate. It is recommended to select all three check boxes (for identification of websites, identification of email users and for identification of software). After that, click **OK**.
5) On the list of trusted certificates, a new section—*DrWeb*—appears. This section contains the added certificate (*SpIDer Gate Trusted Root Certificate* by default).
6) Close the window with the list of certificates by clicking **OK** and then close the page with browser settings (by closing the corresponding tab on the browser tab bar).

Restart the Mozilla Thunderbird mail client.

1) Select **Preferences** on the main menu; in the appeared settings window, click **Advanced**. On the appeared page, select **Certificates**.
2) Click **View Certificates**. In the appeared window, open the **Authorities** tab and click **Import**.
3) In the appeared window, specify a path to the Dr.Web certificate (by default, its file name is `SpIDer Gate Trusted Root Certificate.pem`) and click **Open**.
4) In the appeared window, use the check boxes to specify a necessary trust level for the certificate. It is recommended to select all three check boxes (for identification of websites, identification of email users and for identification of software). After that, click **OK**.
5) On the list of trusted certificates, a new section—*DrWeb*—appears. This section contains the added certificate (*SpIDer Gate Trusted Root Certificate* by default).
6) Close the window with the list of certificates by clicking **OK** and then close the page with mail client settings by clicking **Close**.
7) Restart the mail client.

Adding Dr.Web Certificate to List of Trusted Certificates via Command Line

Besides the graphical user interface, you can use the command line to add the certificate. To generate it, run the following command (you need to specify a name for saving the certificate in PEM format):

```
$ drweb-ctl certificate > <cert_name>.pem
```

After that, add the certificate to the system storage. Different GNU/Linux distributions require different commands to perform this operation.

- On Ubuntu, Debian and Mint:

```
# cp <cert_name>.pem /etc/ssl/certs/
# c_rehash
```

- On CentOS and Fedora:

```
# cp <cert_name>.pem /etc/pki/ca-trust/source/anchors/
# update-ca-trust extract
```

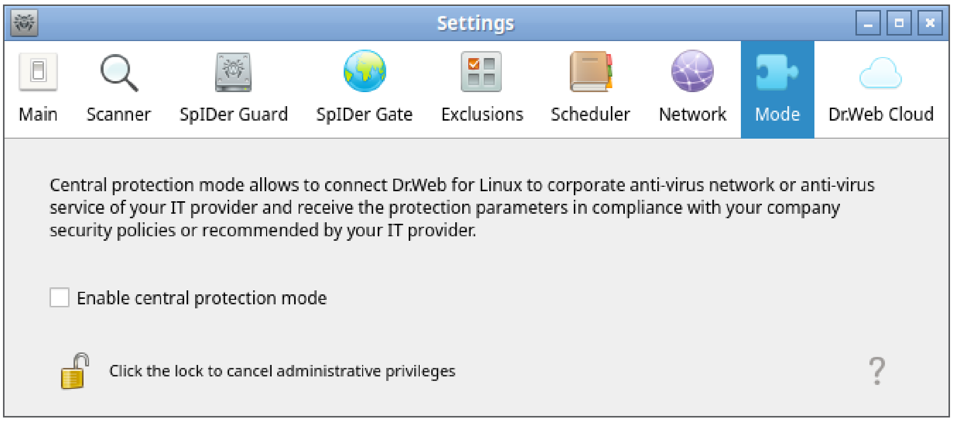
8.1.9.8. Mode Settings

In this section:

- General Information.
- Connection to a centralized protection server.
- Advanced Settings.

General Information

On the **Mode** tab, you can connect Dr.Web Security Space to a centralized protection server (by enabling the centralized protection mode) as well as disconnect from the centralized protection server (if so, Dr.Web Security Space will operate in standalone mode).



Central protection mode allows to connect Dr.Web for Linux to corporate anti-virus network or anti-virus service of your IT provider and receive the protection parameters in compliance with your company security policies or recommended by your IT provider.

□ Enable central protection mode

Click the lock to cancel administrative privileges

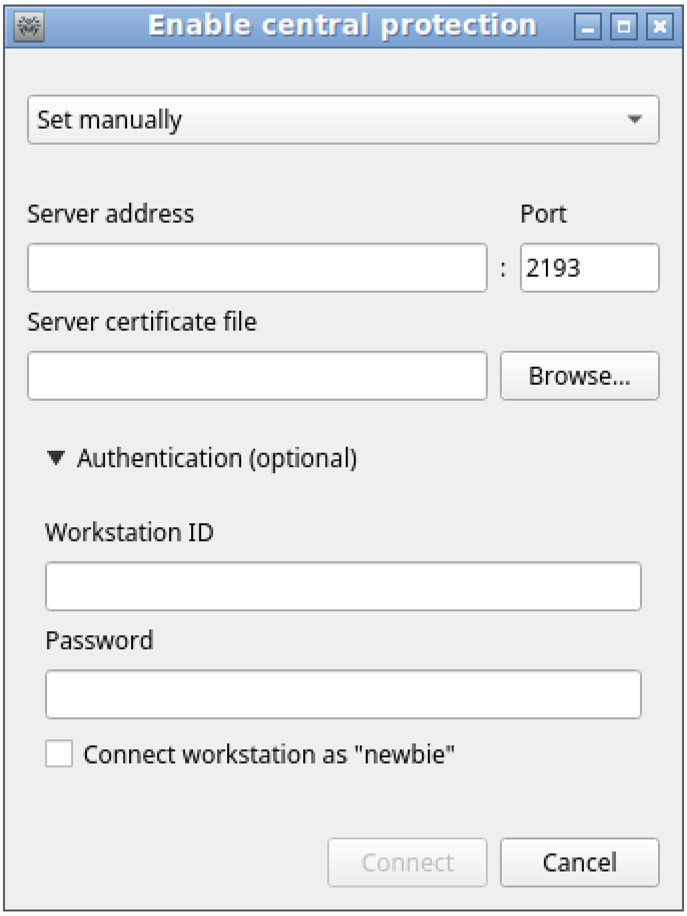
Figure 49. Mode management

To connect Dr.Web Security Space to the centralized protection server or disconnect from it, select or clear the corresponding check box.

> To connect Dr.Web Security Space to the centralized protection server or disconnect from it, the application must have elevated privileges (refer to Managing Application Privileges).

Connection to a Centralized Protection Server

On attempt to establish connection to the centralized protection server, a window with connection parameters appears:



Enable central protection

Set manually

Server address:

Port: 2193

Server certificate file:

▼ Authentication (optional)

Workstation ID:

Password:

□ Connect workstation as "newbie"

Figure 50. Connection to the centralized protection server

Select one of the methods for connecting to the server from the drop-down list located at the top of the window. Three methods are available:

- *Load from file*.
- *Set manually*.
- *Detect automatically*.

If you select the *Load from file* item, specify a path to a server connection settings file provided by an anti-virus network administrator in the corresponding field. If you select *Set manually* or *Detect automatically*, specify an address and a port for connecting to the centralized protection server as well as a path to a certificate file (usually provided by your anti-virus network administrator or internet service provider).

Additionally, in the **Authentication** section you can specify a workstation identifier and a password for authentication on the server, if you know them. If these fields are filled in, then your connection to the server will succeed only if a correct identifier/password pair was entered. If you leave these fields empty, connection to the server is established only if it is approved by the server (either automatically or by the anti-virus network administrator, depending on the server settings).

Furthermore, you can select the **Connect to workstation as newbie** check box. If the newbie option is allowed on the server and the connection is approved, the server automatically generates a unique identifier/password pair, which will be further used for connecting your computer to this server.

> If you connect as a newbie, the centralized protection server generates a new account for your computer even if it already had an account on this server.
>
> Specify connection parameters in strict accordance with the instructions provided by your anti-virus network administrator or service provider.

To connect to the server, specify all of the parameters, click **Connect** and wait for the connection to be established. To close the window without establishing the connection to the server, click **Cancel**.

> Once you have connected Dr.Web Security Space to the centralized protection server, the server will control the application until you put the application in standalone mode. The connection to the server will be established automatically when the operating system starts up (see the Operation Modes section).
>
> If the centralized protection server does not allow the user to start scanning, the page for starting scanning and the **Scanner** button on the Dr.Web Security Space window will be disabled. Furthermore, in this case Scanner will not start scheduled scans.

Advanced Settings

The drop-down list **Maximum storage time for server messages** allows you to select the maximum storage time for the messages on the anti-virus network status and events that are received by your workstation from the centralized protection server to which Dr.Web Security Space is connected. The messages will be automatically deleted at the end of the indicated period even if they are not read.

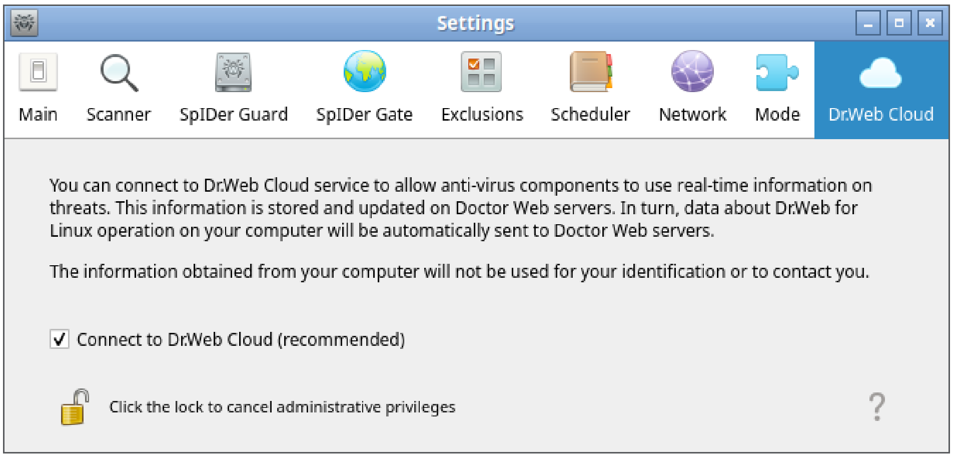
> The messages on the status and events of the anti-virus network can be received only if the anti-virus network administrator has configured message delivery to your workstation on the centralized protection server to which Dr.Web Security Space is connected. Otherwise, the messages cannot be viewed and the drop-down list **Maximum storage time for server messages** is not displayed on the mode settings page.

8.1.9.9. Configuring Dr.Web Cloud

On the **Dr.Web Cloud** tab, you can allow or prohibit Dr.Web Security Space to use Dr.Web Cloud service.

Dr.Web Cloud provides most recent information on threats which is updated on Doctor Web servers in real-time mode and used for anti-virus protection. Depending on update settings, information on threats used by anti-virus components may become out of date. Using of

Dr.Web Cloud can reliably prevent users from viewing unwanted websites and protect your system from infected files.



You can connect to Dr.Web Cloud service to allow anti-virus components to use real-time information on threats. This information is stored and updated on Doctor Web servers. In turn, data about Dr.Web for Linux operation on your computer will be automatically sent to Doctor Web servers.

The information obtained from your computer will not be used for your identification or to contact you.

☑ Connect to Dr.Web Cloud (recommended)

Click the lock to cancel administrative privileges

Figure 51. Dr.Web Cloud tab

To allow or prohibit Dr.Web Security Space to use Dr.Web Cloud service, select or clear the corresponding check box.

For interaction with Dr.Web Cloud service, it is necessary to have an active internet connection.

To allow or prohibit Dr.Web Security Space using of Dr.Web Cloud, the application must have elevated privileges. Refer to Managing Application Privileges.

8.1.10. Additional Information

8.1.10.1. Command-Line Arguments

To start the graphical management interface of Dr.Web Security Space from the command line of the operating system, run the following command:

```
$ drweb-gui [<path>[ <path> …] | <parameters>]
```

where <*path*> is a path to be scanned. You can specify a list of space-separated paths.

The command accepts the following parameters (<*parameters*>):

- `--help (-h)`—display the information about supported command-line parameters and close the graphical management interface.
- `--version (-v)`—display the information about the version of the graphical management interface.
- `--Autonomous (-a)`—run the Dr.Web Security Space graphical management interface in autonomous instance mode.

- `--FullScan`—start the full scan task upon starting the Dr.Web Security Space graphical management interface.
- `--ExpressScan`—start the express scan task upon starting the Dr.Web Security Space graphical management interface.
- `--CustomScan`—start the custom scan task upon starting the Dr.Web Security Space graphical management interface (open page for selection of objects to be scanned).

Example:

```
$ drweb-gui /home/user/
```

This command starts the Dr.Web Security Space graphical management interface, then Scanner starts scanning the files for the specified path (the corresponding task appears in the list of current scans).

8.1.10.2. Starting the Autonomous Instance

Dr.Web Security Space can be run in special mode—as an *autonomous instance*.

If the graphical management interface of Dr.Web Security Space is run as autonomous instance, then it will work with a separate set of service components (the *configuration daemon of Dr.Web Security Space* working in background (`drweb-configd`), Scanner and the scanning engine) run specifically for supporting the running instance of the software.

Aspects of the Dr.Web Security Space graphical management interface running as an autonomous instance:

- To run the Dr.Web Security Space graphical management interface as an autonomous instance, you will need a valid key file, operation in centralized protection mode is not supported (it is possible to install the key file exported from a centralized protection server). In this case, even if Dr.Web Security Space is connected to the centralized protection server, the autonomous instance *does not notify* the centralized protection server of the threats detected in the autonomous instance mode.
- All additional components that are run to serve the work of the autonomous copy of the graphical management interface, will be launched as the current user and will work with a configuration file, separately generated for this session.
- All temporary files and UNIX sockets used for interaction of components are created only in a directory with an unique name, which is created when the autonomous instance is started. The unique temporary directory is created in the directory for temporary files (specified by the `TMPDIR` environment variable).
- The autonomous instance of the graphical management interface *does not start* the SpIDer Guard and SpIDer Gate monitors, only file scanning and quarantine management functions supported by Scanner are available.
- All the required paths (to virus databases, scanning engine and executable files of service components) are set to default values or retrieved from custom environment variables.

- The number of simultaneously running autonomous copies of the graphical management interface is unlimited.
- When the autonomous copy of the graphical management interface is shut down, the set of servicing components is also terminated.

8.2. Working from Command Line

To manage operation of Dr.Web Security Space from the command line, use the `drweb-ctl` utility of the Dr.Web Ctl component. The operating principles and usage examples of the utility are described in detail in the respective sections.

9. Dr.Web Security Space Components

This section contains a description of the Dr.Web Security Space components. For each of them, you can find information about its functions, operation principles and parameters stored in the configuration file.

9.1. Dr.Web ConfigD

The Dr.Web ConfigD configuration management daemon is the main control component of Dr.Web Security Space. It provides centralized storage for settings of all Dr.Web Security Space components, manages the operation of all components and organizes trusted data exchange among them.

Dr.Web ConfigD performs the following functions:

- starting and stopping the components of Dr.Web Security Space depending on settings;
- restarting the components automatically in case of failures;
- starting the components upon the request of other components;
- informing the components of changed settings;
- enabling the centralized management of configuration parameters;
- passing the information from the key file in use to the components;
- receiving the license information from the components;
- receiving the new license information from the specialized components;
- informing the running components of license changes.

9.1.1. Operating Principles

The Dr.Web ConfigD configuration management daemon always runs with superuser privileges (*root*). It starts other Dr.Web Security Space components and communicates with them via a preliminarily open socket. The configuration daemon accepts connections from other Dr.Web Security Space components via an information socket (publicly accessible) and an administrative socket (accessible only to the components running with superuser privileges). The daemon loads configuration parameters and license information from files or receives this information from a centralized protection server via Dr.Web ES Agent and sets valid default values of the configuration parameters. By the time any component starts or receives the `SIGHUP` signal, the configuration management daemon has a comprehensive and consistent set of configuration parameters for all Dr.Web Security Space components.

Upon receiving the `SIGHUP` signal, Dr.Web ConfigD reloads the configuration parameters and license information. If required, the daemon also instructs all components to reload their configuration parameters.

Upon receiving the `SIGTERM` signal, Dr.Web ConfigD shuts down all components and then finishes its own operation. Dr.Web ConfigD removes all temporary files of the components after they are shut down.

Principles of Interaction with Other Components

1. All components receive the configuration parameters and license information from Dr.Web ConfigD at startup. Only these settings are used by the components in their further operation.
2. Dr.Web ConfigD forwards messages from all components started with it to a unified log. All messages output to *stderr* by the components are gathered by Dr.Web ConfigD and stored in the unified log of Dr.Web Security Space with a mark indicating the component that reported an error and time of its occurrence.
3. Upon shutting down, all controlled components return an exit code. If the code differs from `101`, `102` and `103`, the component will be restarted and the corresponding message from *stderr* will be output to the Dr.Web Security Space log.
    - Code 101 is returned when the component cannot operate with the current license. The component will be restarted only after the license parameters are modified.
    - Code 102 is returned when the component cannot operate with the current configuration parameters. If some configuration parameters were modified, Dr.Web ConfigD will try to restart the component.
    - Code `103` is returned in case the components started by Dr.Web ConfigD upon request (Dr.Web Scanning Engine and Dr.Web File Checker) have been idle for a long time. The timeout after which the component is shut down with error code `103` is specified in the settings of the corresponding component (the `IdleTimeLimit` parameter).
    - If new configuration parameters received from Dr.Web ConfigD by the component cannot be applied on the fly, the component exits with code `0` so that Dr.Web ConfigD can restart it.
    - If the component cannot connect to Dr.Web ConfigD or a communication protocol error occurs, the component outputs a corresponding message to *stderr* and exits with code `1`.
4. Signal exchange is maintained.
    - Dr.Web ConfigD sends the `SIGHUP` signal to the component instructing it to apply the modified configuration parameters.
    - Dr.Web ConfigD sends the `SIGTERM` signal to the component instructing it to shut down. After receiving the signal, the component should shut down within 30 seconds.
    - If the component does not shut down within 30 seconds, Dr.Web ConfigD sends the `SIGKILL` signal to forcibly shut down the component.

9.1.2. Command-Line Arguments

To start the Dr.Web ConfigD configuration management daemon from the command line, run the following command:

```
$ /opt/drweb.com/bin/drweb-configd [<parameters>]
```

Dr.Web ConfigD accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.
`--config`	Function: Use the specified configuration file for further operation. Short form: `-c` Arguments: <*path to file*>—path to the configuration file to be used.
`--daemonize`	Function: Run the component as a daemon; that is, without access to the terminal. Short form: `-d` Arguments: None.
`--pid-file`	Function: Use the specified PID file for further operation. Short form: `-p` Arguments: <*path to file*>—path to the file to store the process identifier (PID).

Example:

```
$ /opt/drweb.com/bin/drweb-configd -d -c /etc/opt/drweb.com/drweb.ini
```

This command runs Dr.Web ConfigD as a daemon, which forces the component to use the configuration file `/etc/opt/drweb.com/drweb.ini`.

Startup Notes

To enable the operation of Dr.Web Security Space, the component must run as a daemon. Under normal conditions, Dr.Web ConfigD starts automatically at the startup of the operating system; for this purpose the component is supplied with the `drweb-configd` management script located in a system directory (`/etc/init.d`). To manage the operation of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line (the tool is run with the `drweb-ctl` command).

> To get documentation for this component from the command line, run the `man 1 drweb-configd` command.

9.1.3. Configuration Parameters

The Dr.Web ConfigD daemon uses configuration parameters specified in the `[Root]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`DefaultLogLevel` *{logging level}*	Default event logging level *for all* Dr.Web Security Space components. The value of this parameter is used if a custom logging level is not specified for some component. Default value: `Notice`
`LogLevel` *{logging level}*	Logging level of the Dr.Web ConfigD component. Default value: `Notice`
`Log` *{log type}*	Logging method of the configuration daemon and of those components for which another value of this parameter is not specified. Upon its initial startup, before the configuration file is read, the configuration daemon uses the following values of the parameter: • as a daemon (if run with the `-d` option)—`SYSLOG:Daemon`; • `Stderr` otherwise. If a component is operating in the background (was started with the `-d` option from the command line), the `Stderr` value *cannot* be used for this parameter. Default value: `SYSLOG:Daemon`
`PublicSocketPath` *{path to file}*	Path to the socket used for interaction by all Dr.Web Security Space components. Default value: `/var/run/.com.drweb.public`
`AdminSocketPath` *{path to file}*	Path to the socket used for interaction by Dr.Web Security Space components with elevated (administrative) privileges. Default value: `/var/run/.com.drweb.admin`
`CoreEnginePath` *{path to file}*	Path to the dynamic library of Dr.Web Virus-Finding Engine.

Parameter	Description
	Default value: `/var/opt/drweb.com/lib/drweb32.dll`
`VirusBaseDir` *{path to directory}*	Path to the directory with virus database files. Default value: `/var/opt/drweb.com/bases`
`KeyPath` *{path to file}*	Path to the key file (license or demo). Default value: `/etc/opt/drweb.com/drweb32.key`
`CacheDir` *{path to directory}*	Path to the cache directory (being used to store cache for updates as well as cache for information about scanned files). Default value: `/var/opt/drweb.com/cache`
`TempDir` *{path to directory}*	Path to the directory with temporary files. Default value: *Path copied from the system environment variable* `TMPDIR`, `TMP`, `TEMP` or `TEMPDIR` (the environment variables are searched in this particular order). Otherwise `/tmp`, if none of them was found.
`RunDir` *{path to directory}*	Path to the directory with all PID files of running components and sockets used for interaction by Dr.Web Security Space components. Default value: `/var/run`
`VarLibDir` *{path to directory}*	Path to the directory with libraries used by Dr.Web Security Space components. Default value: `/var/opt/drweb.com/lib`
`VersionDir` *{path to directory}*	Path to a directory, where the information about the current versions of Dr.Web Security Space components is stored. Default value: `/var/opt/drweb.com/version`
`DwsDir` *{path to directory}*	Path to the directory that contains files of an automatically updated database of web resource categories. Default value: `/var/opt/drweb.com/dws`
`AdminGroup` *{group name | GID}*	Group of users with administrative privileges for Dr.Web Security Space management. These users, in addition to the superuser (the *root* user), are allowed to elevate privileges of Dr.Web Security Space components to superuser privileges. Default value: *Defined automatically* during Dr.Web Security Space installation
`TrustedGroup` *{group name | GID}*	Group of trusted users. The parameter is used by the SpIDer Gate network traffic monitoring component. Network traffic of such users is skipped by SpIDer Gate. Default value: `drweb`
`DebugIpc` *{logical}*	Log or do not log detailed IPC messages at the debug level (with `LogLevel = DEBUG`). These messages reflect interaction of the

Parameter	Description
	configuration management daemon with other components. Default value: `No`
`UseCloud` *{logical}*	Use or do not use the Dr.Web Cloud service to receive information about whether files and URLs are malicious or not. Default value: `No`
`AntispamDir` *{path to directory}*	Path to the directory that contains the files used by the antispam library. Default value: `/var/opt/drweb.com/antispam`
`VersionNotification` *{logical}*	Notify or do not notify the user of updates for the current version of Dr.Web Security Space. Default value: `Yes`
`UseVxcube` *{logical}*	Use or do not use Dr.Web vxCube to analyze email attachments as an external filter connected to the MTA. Default value: `No`
`VxcubeApiAddress` *{string}*	Domain name (FQDN) or IP address of a host running the Dr.Web vxCube API server to be connected to. Default value: *(not specified)*
`VxcubeApiKey` *{string}*	Dr.Web vxCube API key. Default value: *(not specified)*
`VxcubeProxyUrl` *{connection address}*	Address of the proxy server used for connecting to Dr.Web vxCube. Only HTTP proxies without authorization are supported. Possible values: <*connection address*>—proxy server connection parameters in the following format: `http://`<*host*>`:`<*port*>, where: • <*host*> is the host address of the proxy server (an IP address or a domain name, i.e. FQDN); • <*port*> is the port to be used. Default value: *(not specified)*

9.2. Dr.Web Ctl

In this section

- General Information
- Remote host scanning

General Information

You can manage operation of Dr.Web Security Space from the command line of the operating system. For that, you can use the special Dr.Web Ctl utility (`drweb-ctl`). You can use it to perform the following operations:

- Start scanning file system objects including boot records.
- Launch of scanning of files on remote network hosts (see note below).
- Start updating anti-virus components (virus databases, the scan engine, and so on depending on the distribution).
- View and change parameters of the Dr.Web Security Space configuration.
- View the status of the Dr.Web Security Space components and statistics on detected threats.
- Connect to the centralized protection server or disconnect from it.
- View quarantine and manage quarantined objects (via the Dr.Web File Checker component).
- Connect to the centralized protection server or disconnect from it.

User commands to control Dr.Web Security Space will only take effect if the Dr.Web ConfigD configuration daemon is running (by default, it is automatically run on system startup).

> Note that some control commands require superuser privileges.
>
> To elevate privileges, use the `su` command (change the current user) or the `sudo` command (run the specified command as another user).

The `drweb-ctl` tool supports auto-completion of commands for managing Dr.Web Security Space operation if this option is enabled your command shell. If the command shell does not allow auto-completion, you can configure this option. For that purpose, refer to the instruction manual for the used OS distribution.

> When shutting down, the tool returns the exit code according to convention for the POSIX compliant systems: 0 (zero)—if an operation is successfully completed, non-zero—if otherwise.
>
> Note that the tool only returns a non-null exit code in the case of internal error (for example, the tool could not connect to a component, the requested operation could not be executed, and so on). If the tool detects and possibly neutralizes a threat, it returns the null exit code, because the requested operation (such as `scan`, and so on) is successfully completed. If you need to define the list of the detected threats and applied actions, analyze the messages displayed on the console.
>
> Codes of all errors are listed in the Appendix G. Known Errors section.

Remote host scanning

Dr.Web Security Space allows you to scan files located on remote network hosts for threats. Such hosts can be not only fully-featured computing machines, such as workstations and servers, but also routers, set-top boxes, and other smart devices of the Internet of Things. To perform the remote scanning, the remote host has to provide a remote terminal access via *SSH* (*Secure Shell*) or *Telnet*. To access the device, you need to know an IP address and a domain name of the remote host, as well as the credentials of the user that can remotely access the system via *SSH* or *Telnet*. This user must have access rights to the scanned files (at least the reading rights).

This function can be used only for detection of malicious and suspicious files on a remote host. Elimination of threats (i.e. isolation in the quarantine, removal, and cure of malicious objects) using remote scanning is impossible. To eliminate the detected threats on the remote host, use administration tools provided directly by this host. For example, for routers and other smart devices, update the firmware; for computing machines, establish a connection (in a remote terminal mode, as one of the options) and perform the respective operations in the file system (remove or move files, etc.), or run the anti-virus software installed on them.

Remote scanning is only performed via the command-line tool `drweb-ctl` (using the command `remotescan`).

9.2.1. Command-Line Call Format

1. Command Format for Calling the Command-Line Utility to Manage the Product

The call format for the command-line tool which manages Dr.Web Security Space operation is as follows:

```
$ drweb-ctl [<general options> | <command> [<argument>] [<command options>]]
```

where:

- *<general options>*—options that can be applied on startup when the command is not specified or can be applied for any command. Not mandatory for startup.
- *<command>*—command to be run by Dr.Web Security Space (for example, start scanning, output the list of quarantined objects and so on).
- *<argument>*—command argument. Depends on the specified command. Some commands do not accept arguments.
- *<command options>*—options for managing the operation of the specified command. Depend on the command. Some commands do not accept options.

2. General Options

The following general options are available:

Option	Description
`-h, --help`	Show general help information and exit. To display the help information on any command, use the following call: `$ drweb-ctl <command> -h`
`-v, --version`	Show the module version and exit
`-d, --debug`	Show debug information when running the specified command. It cannot be run if a command is not specified. Use the following call: `$ drweb-ctl <command> -d`

3. Commands

Commands to manage Dr.Web Security Space can be separated into the following groups:

- anti-virus scanning commands;
- commands to manage updates and operation in centralized protection mode;
- configuration management commands;
- commands to manage detected threats and quarantine;
- information commands.

> To get documentation for this component from the command line, run the `man 1 drweb-ctl` command.

3.1. Anti-virus Scanning Commands

The following commands to manage anti-virus scanning are available:

Command	Description
`scan <path>`	**Purpose:** Initiate scanning the specified file or directory by the file scanning component Dr.Web File Checker. **Arguments** *<path>*—path (can be relative) to the file or directory to be scanned.

Command	Description
	This argument may be omitted if you use the `--stdin` or the `--stdin0` option. To specify several files that satisfy a certain criterion, use the `find` utility (see Usage Examples) and the `--stdin` or `--stdin0` option. **Options** `-a [--Autonomous]`—run standalone instances of Dr.Web Scanning Engine and Dr.Web File Checker to perform the specified scan, shutting them down after it is completed. Threats detected during standalone scanning will not be added to the common list of threats detected displayed by the `threats` command (see below), and a centralized protection server will not be notified of them, if Dr.Web Security Space is controlled by it. `--stdin`—get the list of paths to be scanned from the standard input stream (*stdin*). Paths in the list must be separated with the new line character (`\n`). `--stdin0`—get the list of paths to scan from the standard input string (*stdin*). Paths in the list must be separated by the zero character NUL (`\0`). When using `--stdin` and `--stdin0` options, the paths on the list should not contain patterns or regular expressions for a search. We recomment that you use the `--stdin` and `--stdin0` options to process a path list generated by an external utility, for example, `find` in the `scan` command (see Usage Examples). `--Exclude` <*path*>—excluded path. The path can be relative and contain a file mask (with the following wildcards: `?` and `*`, as well as character classes `[ ]`, `[! ]` and `[^ ]`). Optional parameter; can be set more than once. `--Report` <*type*>—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` <*time interval*>—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`.

Command	Description
	`--PackerMaxLevel` <*number*>—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ArchiveMaxLevel` <*number*>—maximum nesting level while scanning archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MailMaxLevel` <*number*>—maximum nesting level while scanning files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MaxCompressionRatio` <*ratio*>—maximum compression ratio of scanned objects. The value must be no less than `2`. Default value: `3000`. `--MaxSizeToExtract` <*number*>—maximum size for files enclosed in archives. Files which size is greater than the value of this parameter will be skipped when scanning. The size is specified as a number with a suffix (*b, kb, mb, gb*). If no suffix is specified, the value is treated as a size in bytes. Default value: *(not specified)*. `--HeuristicAnalysis` <*On|Off*>—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--OnKnownVirus` <*action*>—action to perform upon detection of a known threat by using the signature-based analysis.

Command	Description
	Possible actions: REPORT, CURE, QUARANTINE, DELETE. Default value: REPORT. --OnIncurable <*action*>—action to perform upon detection an incurable threat or when the curing action (CURE) has failed. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnSuspicious <*action*>—action to perform upon detection of a suspicious object using the heuristic analysis. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnAdware <*action*>—action to perform upon detection of adware. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnDialers <*action*>—action to perform upon detection of a dialer. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnJokes <*action*>—action to perform upon detection of joke software. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnRiskware <*action*>—action to perform upon detection of riskware. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. --OnHacktools <*action*>—action to perform upon detection of a hacktool. Possible actions: REPORT, QUARANTINE, DELETE. Default value: REPORT. If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined (QUARANTINE) and not deleted (DELETE). --FollowSymlinks—resolve symlinks automatically
bootscan <*device*> | ALL	**Purpose:** Start scanning boot records on specified disks using the file scan component Dr.Web File Checker. Both MBR and VBR records are scanned. **Arguments** <*disk drive*>—path to the block file of a disk device whose boot record you want to scan. You can specify several disk devices separated by spaces.

Command	Description
	The argument is mandatory. If `ALL` is specified instead of the device file, all boot records on all available disk devices will be checked. **Options** `-a [--Autonomous]`—run standalone instances of Dr.Web Scanning Engine and Dr.Web File Checker to perform the specified scan, shutting them down after it is completed. ⚠ Threats detected during standalone scanning will not be added to the common list of threats detected displayed by the `threats` command (see below), and a centralized protection server will not be notified of them, if Dr.Web Security Space is controlled by it. `--Report` *<type>*—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` *<time interval>*—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--HeuristicAnalysis` *<On|Off>*—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--Cure` *<Yes|No>*—enable or disable attempts to cure detected threats. If the value is set to `No`, only a notification about a detected threat is displayed. Default value: `No`. `--ShellTrace`—display additional debug information when scanning a boot record
`procscan`	**Purpose:** Initiate scanning of executables containing the code of currently running system processes with the Dr.Web File Checker component. If a malicious executable file is detected, it is neutralized, and all processes run by this file are forced to terminate. **Arguments**: None.

Command	Description
	Options `-a [--Autonomous]`—run standalone instances of Dr.Web Scanning Engine and Dr.Web File Checker to perform the specified scan, shutting them down after it is completed. ⚠ Threats detected during standalone scanning will not be added to the common list of threats detected displayed by the `threats` command (see below), and a centralized protection server will not be notified of them, if Dr.Web Security Space is controlled by it. `--Report` <*type*>—type of the scan report. Allowed values: - `BRIEF`—brief report; - `DEBUG`—detailed report; - `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` <*time interval*>—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--PackerMaxLevel` <*number*>—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--HeuristicAnalysis` <*On|Off*>—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--Exclude` <*path*>—path to be excluded from scanning. The path can contain a file mask with the following allowed symbols: `?` and `*`, as well as the symbol classes `[ ]`, `[! ]`, `[^ ]`. The path (including the path with the file mask) must be absolute.

Command	Description
	Optional parameter; can be set more than once. `--OnKnownVirus` <*action*>—action to perform upon detection of a known threat by using the signature-based analysis. Possible actions: `REPORT`, `CURE`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnIncurable` <*action*>—action to perform upon detection an incurable threat or when the curing action (`CURE`) has failed. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnSuspicious` <*action*>—action to perform upon detection of a suspicious object using the heuristic analysis. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnAdware` <*action*>—action to perform upon detection of adware. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnDialers` <action>—action to perform upon detection of a dialer. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnJokes` <*action*>—action to perform upon detection of joke software. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnRiskware` <*action*>—action to perform upon detection of riskware. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnHacktools` <*action*>—action to perform upon detection of a hacktool. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. If a threat is detected in an executable file, Dr.Web Security Space terminates all processes started by the file.
`netscan [`<*path*>`]`	**Purpose:** Start distributed scanning of the specified file or directory using the Dr.Web Network Checker agent for network data scanning. If there are no configured connections to other hosts that are running Dr.Web for UNIX, then the scanning will be done only using the locally available scan engine (similar to the `scan` command).

Command	Description
	Arguments *<path>*—path to the file or directory to be scanned. If this argument is omitted, data from the stdin input stream will be scanned. **Options** `--Report` *<type>*—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` *<time interval>*—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--HeuristicAnalysis` *<On|Off>*—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--PackerMaxLevel` *<number>*—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ArchiveMaxLevel` *<number>*—maximum nesting level while scanning archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MailMaxLevel` *<number>*—maximum nesting level while scanning files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`.

Command	Description
	`--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MaxCompressionRatio` <*ratio*>—maximum compression ratio of scanned objects. The value must be no less than `2`. Default value: `3000`. `--MaxSizeToExtract` <*number*>—maximum size for files enclosed in archives. Files which size is greater than the value of this parameter will be skipped when scanning. The size is specified as a number with a suffix (*b, kb, mb, gb*). If no suffix is specified, the value is treated as a size in bytes. Default value: *(not specified)*. `--Cure` <*Yes*|*No*>—enable or disable attempts to cure detected threats. If the value is set to `No`, only a notification about a detected threat is displayed. Default value: `No`
`flowscan` <*path*>	**Purpose:** Start scanning the specified file or directory via Dr.Web File Checker using the “flow” method (normally this method is used internally by SpIDer Guard). ⚠ For on-demand scanning of files and directories, it is recommended that you use the `scan` command. **Arguments** <*path*>—path to the file or directory to be scanned. **Options** `--ScanTimeout` <*time interval*>—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--HeuristicAnalysis` <*On*|*Off*>—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--PackerMaxLevel` <*number*>—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also

Command	Description
	include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ArchiveMaxLevel` <*number*>—maximum nesting level while scanning archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MailMaxLevel` <*number*>—maximum nesting level while scanning files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MaxCompressionRatio` <*ratio*>—maximum compression ratio of scanned objects. Must be no less than `2`. Default value: `3000`. `--OnKnownVirus` <*action*>—action to perform upon detection of a known threat by using the signature-based analysis. Possible actions: `REPORT`, `CURE`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnIncurable` <*action*>—action to perform upon detection an incurable threat or when the curing action (`CURE`) has failed. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnSuspicious` <*action*>—action to perform upon detection of a suspicious object using the heuristic analysis. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnAdware` <*action*>—action to perform upon detection of adware.

Command	Description
	Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnDialers` <*action*>—action to perform upon detection of a dialer. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnJokes` <*action*>—action to perform upon detection of joke software. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnRiskware` <*action*>—action to perform upon detection of riskware. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. `--OnHacktools` <*action*>—action to perform upon detection of a hacktool. Possible actions: `REPORT`, `QUARANTINE`, `DELETE`. Default value: `REPORT`. If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined (`QUARANTINE`) and not deleted (`DELETE`).
`rawscan` <*path*>	**Purpose**: Start “raw” scanning of the specified file or directory with Dr.Web Scanning Engine directly, without the use of Dr.Web File Checker. Threats detected during “raw” scanning are not included in the list of detected threats that can be displayed with the `threats` command (see below). It is recommended that you use this command only to debug the functioning of Dr.Web Scanning Engine. Note that the command outputs the “cured” status, if at least *one* threat is neutralized of those threats that are detected in a file (not *all* threats might be neutralized). Thus, it is *not recommended* to use this command if you need thorough file scanning. In the latter case it is recommended to use the scan`scan` command. **Arguments** <*path*>—path to the file or directory to be scanned.

Command	Description
	Options `--ScanEngine` <*path*>—path to the UNIX socket of Dr.Web Scanning Engine. If not specified, a standalone instance of the scan engine will be started (which will be shut down once the scanning is complete). `--Report` <*type*>—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` <*time interval*>—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--PackerMaxLevel` <*number*>—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ArchiveMaxLevel` <*number*>—maximum nesting level while scanning archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MailMaxLevel` <*number*>—maximum nesting level while scanning files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped.

Command	Description
	Default value: `8`. `--MaxCompressionRatio` <*ratio*>—maximum compression ratio of scanned objects. Must be no less than `2`. Default value: `3000`. `--MaxSizeToExtract` <*number*>—maximum size for files enclosed in archives. Files which size is greater than the value of this parameter will be skipped when scanning. The size is specified as a number with a suffix (*b, kb, mb, gb*). If no suffix is specified, the value is treated as a size in bytes. Default value: *(not specified)*. `--HeuristicAnalysis` <*On|Off*>—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--Cure` <*Yes|No*>—enable or disable attempts to cure detected threats. If the value is set to `No`, only a notification about a detected threat is displayed. Default value: `No`. `--ListCleanItem`—output the list of clean (non-infected) files found inside the container that was scanned. `--ShellTrace`—enable display of additional debug information when scanning a file. `--Output` <*path to file*>—duplicate the output of the command to the specified file
`remotescan` <*host*> <*path*>	**Purpose**: Start scanning the specified file or directory at the specified remote host having connected to it using *SSH* or *Telnet*. ⚠ Threats detected during remote scanning are not neutralized and are also not added to the list of detected threats displayed with the `threats` command (see below). This function can be used only for detection of malicious and suspicious files on a remote host. To eliminate detected threats on the remote host, it is necessary to use administration tools provided directly by this host. For example, for routers, set-top boxes, and other “smart” devices, a mechanism for a firmware update can be used; for computing machines, it can be done by connecting to them (as an option, using a remote terminal mode) and by performing corresponding operations in their file system (file removal or moving and so on), or by running an anti-virus software installed on them.

Command	Description
	Arguments • <*host*>—IP address or a domain name of the remote host to be connected to for scanning. • <*path*>—path to the file or directory to be scanned (the path must be absolute). **Options** `-l [--Login]` <*name*>—login (user name) used for authorization on the remote host via the selected protocol. If a user name is not specified, an attempt is made to connect to a remote host as the user who started the command. `-i [--Identity]` <*path to file*>—private key file used for authentication of the specified user via the selected protocol. `-m [--Method]` <*SSH*|*Telnet*>—remote host connection method (protocol). If the method is not specified, SSH is used. `-p [--Port]` <*number*>—number of the port on the remote host for connecting via the selected protocol. Default value: Default port for the selected protocol (`22` for SSH, `23` for Telnet). `--UseChannels` <*number*>—number of data transfer channels. Default value: `5`. `--Password` <*password*>—password used for authentication of a user via the selected protocol. ⚠ Password is passed as plain text. `--Report` <*type*>—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `--ScanTimeout` <*time interval*>—timeout for scanning one file in milliseconds. If the value is set to `0`, scanning time is not limited. Default value: `0`. `--PackerMaxLevel` <*number*>—maximum nesting level while scanning packed objects. A packed object is executable code compressed with specialized software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects which may also

Command	Description
	include packed objects and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ArchiveMaxLevel` <*number*>—maximum nesting level while scanning archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MailMaxLevel` <*number*>—maximum nesting level while scanning files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--ContainerMaxLevel` <*number*>—maximum nesting level while scanning other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. If the value is set to `0`, nested objects are skipped. Default value: `8`. `--MaxCompressionRatio` <*ratio*>—maximum compression ratio of scanned objects. Must be no less than `2`. Default value: `3000`. `--MaxSizeToExtract` <*number*>—maximum size for files enclosed in archives. Files which size is greater than the value of this parameter will be skipped when scanning. The size is specified as a number with a suffix (*b, kb, mb, gb*). If no suffix is specified, the value is treated as a size in bytes. Default value: *(not specified)*. `--HeuristicAnalysis` <*On|Off*>—enable or disable the heuristic analysis during the scanning. Default value: `On`. `--Exclude` <*path*>—path to be excluded from scanning. The path can contain a file mask with the following allowed symbols: `?` and `*`, as well as the symbol classes `[ ]`, `[! ]`, `[^ ]`. The path (including the path with the file mask) must be absolute. Optional parameter; can be set more than once.

Command	Description
	`--TransferListenAddress` *<address>*—address for receiving files transferred from the remote device for scanning. Optional parameter. If not indicated, an arbitrary address is used. `--TransferListenPort` *<port>*—port for receiving files transferred from the remote device for scanning. Optional parameter. If not indicated, an arbitrary port is used. `--TransferExternalAddress` *<address>*—address for the remote device to send files for scanning. Optional parameter. If not indicated, the `--TransferListenAddress` option value or the outgoing address of the already established session is used. `--TransferExternalPort` *<port>*—port to transfer files for scanning, specified for the remote device. Optional parameter. If not indicated, an automatically determined port is used. `--ForceInteractive`—use the SSH interactive session (only for SSH connections). *Optional parameter.*
`checkmail` *<path to file>*	**Purpose:** Perform scan of an email message saved to a file for threats, signs of spam, malicious links, or non-compliance with rules of mail processing (using the email processing component Dr.Web MailD). The output stream (*stdout*) will contain message scanning results and the action applied to this message after scanning by the email processing component Dr.Web MailD. **Arguments** *<path to file>*—path to the file of the email message to be scanned. Mandatory argument. **Options** `--Report` *<type>*—type of the scan report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. Default value: `BRIEF`. `-r [--Rules]` <list of rules>—list of rules to follow while scanning an email message. If the rules are not indicated, the default set of rules is applied, in particular: `threat_category in (KnownVirus, VirusModification, UnknownVirus, Adware, Dialer) : REJECT`

Command	Description

	`total_spam_score gt 0.80 : REJECT`
	`url_category in (InfectionSource, NotRecommended, CopyrightNotice) : REJECT`
	If Dr.Web Anti-Spam is not installed, the scanning rule for spam (the second string) will be automatically excluded from the set.
	`-c [--Connect]` <*IP*>:<*port*>—network socket to be used as a connection address of a sender of the scanned message.
	`-e [--Helo]` <*name*>—identifier of the client that sent the message (an IP address or FQDN host, as for the `HELO/EHLO` SMTP command).
	`-f [--From]` <*email*>—email address of a sender (as for the `MAIL FROM` SMTP command).
	If the address is not indicated, the respective address from the email message will be used.
	`-t [--Rcpt]` <*email*>—email address of a recipient (as for the `RCPT TO` SMTP command).
	If the address is not indicated, the respective address from the email message will be used.
	If email processing component is not installed, calling this command will return an error.

`mailquarantine`	**Purpose:** Configure the Dr.Web Mail Quarantine service component that manages email message queues.
	Arguments: None.
	Options
	`--Flush`—pass scheduled messages from the specified queue to the queue for immediate processing. Requires the `--Queue` option to be specified. Use the call:
	`$ drweb-ctl mailquarantine --Queue <queue> --Flush`
	`--Show`—display the specified message queue. Requires the `--Queue` option to be specified. Use the following call:
	`$ drweb-ctl mailquarantine --Queue <queue> --Show`
	`--Stat`—display statistics on all message queues;
	`--CheckHealth`—perform a consistency check on the message database;
	`--FixHealth`—fix consistency errors in the message database;

Command	Description
	-q [--Queue] <*queue*>—specify the message queue to be processed. Allowed values: • SmtpFresh—messages to be scanned in the SMTP mode; • SmtpAccepted—messages scanned and accepted in the SMTP mode; • BccFresh—messages to be scanned in the BCC mode; • BccAccepted—messages scanned and accepted in the BCC mode. -l [--Limit] <*number*>—maximum number of displayed messages from the selected queue. -d [--Debug]—display debug information while running the specified command. Useless without specifying the command. Use the following call: `$ drweb-ctl mailquarantine <command> -d`

3.2. Commands to manage updates and operation in the centralized protection mode

The following commands for managing updates are available, as well as commands for operation in the centralized protection mode:

Command	Description
update	**Purpose**: Start the updating process of anti-virus components (virus databases, the scan engine and so on, depending on the distribution) from Doctor Web update servers or a local cloud using Dr.Web MeshD, terminate the updating process if it is already running, or perform rollback of the latest update to the previous versions of updated files. The command has no effect if Dr.Web Security Space is connected to the centralized protection server. **Arguments**: None. **Options** --From <*path*>—apply updates offline from a specified directory. --Path <*path*>—store files for updating offline in a specified directory. If this directory already has files, then they will be updated. --Rollback—roll back the last update and restore the previous version of the files that have been updated during the last update. --Stop—terminate the running update process

Command	Description

`esconnect`
`<`*server*`>[:<`*port*`>]`
	Purpose: Connect Dr.Web Security Space to the specified centralized protection server. For details on the operation modes, refer to the Operation Modes section.
	Arguments
	• <*server*>—IP address or network name of the host on which the centralized protection server is operating. This argument is mandatory.
	• <*port*>—port number used by the centralized protection server. The argument is optional and should be specified only if the centralized protection server uses a non-standard port.
	Options
	`--Certificate` <*path*>—file path to a certificate of the centralized protection server, the connection to which will be established.
	`--Login` <*ID*>—login (workstation identifier) used for connection to the centralized protection server.
	`--Password` <*password*>—password for connection to the centralized protection server.
	`--Compress` <*On|Off*>—enable (`On`) or disable (`Off`) forced compression of transmitted data. If not specified, usage of compression is determined by the server.
	`--Encrypt` <*On|Off*>—enable (`On`) or disable (`Off`) forced encryption of transmitted data. If not specified, usage of encryption is determined by the server.
	`--Newbie`—connect as a "newbie" (get a new account on the server).
	`--Group` <*ID*>—identifier of the group to which the workstation is added on connection.
	`--Rate` <*ID*>—identifier of the tariff group applied to your workstation when it is included in one of the centralized protection server groups (can be specified only together with the `--Group` option).
	`--CfgFile` <*path*>—connect to the centralized protection server using the configuration file with connection settings.
	This command requires `drweb-ctl` to be started with superuser (the *root* user) privileges. If necessary, use the `su` or `sudo` commands.

`esdisconnect`	**Purpose**: Disconnect Dr.Web Security Space from the centralized protection server and switch it to a standalone mode. The command has no effect if Dr.Web Security Space already operates in standalone mode. **Arguments**: None.

Command	Description
	Options: None. This command requires `drweb-ctl` to be started with superuser (the *root* user) privileges. If necessary, use the `su` or `sudo` commands.

3.3. Configuration Management Commands

The following commands to manage configuration are available:

Command	Description
`cfset` *<section>*`.`*<parameter>* *<value>*	**Purpose**: Change the active value of the specified parameter in the current configuration of Dr.Web Security Space. **Arguments** • *<section>*—name of the configuration file section which provides the parameter. This argument is mandatory. • *<parameter>*—name of the parameter to be changed. This argument is mandatory. • *<value>*—new parameter value. This argument is mandatory. To specify a parameter value, the format *<section>*.*<parameter>* *<value>* is always used, the assignment character = is not used. If you want to indicate several parameter values, you need to repeatedly call the `cfset` command, as many times as the number of parameter values you want to add. To add a new value to the list of parameter values, you need to use the `-a` option (see below). You cannot specify the string *<parameter>* *<value 1>*, *<value 2>* as an argument, because the string "*<value 1>*, *<value 2>*" will be considered one value of *<parameter>*. For description of the configuration file, refer to the section Appendix D. Dr.Web Security Space Configuration File, as well as the documentation displayed upon running `man 5 drweb.ini`. **Options** `-a [--Add]`—do not substitute the current parameter value but add the specified value to the list (allowed only for parameters that can accept a list of values). This option should also be used for adding new parameter groups with a tag.

Command	Description
	`-e [--Erase]`—do not substitute the current parameter value but remove the specified value from the list (allowed only for parameters that can have several values, specified as a list). `-r [--Reset]`—reset the parameter value to the default. At that, <*value*> is not required in the command and is ignored if specified. Options are not mandatory. If they are not specified, then the current parameter value (including a list of values) are substituted with the specified value. If you use the `-r` option for sections that contain individualized parameter settings for , the parameter value in the individualized settings section will be changed to the value of its corresponding "parent" parameter in the component settings section. This command requires `drweb-ctl` to be started with superuser (the *root* user) privileges. If necessary, use the `su` or `sudo` commands.
`cfshow [`<*section*>`[.`<*parameter*>`]]`	**Purpose**: Display parameters of the current configuration of Dr.Web Security Space. The command to display parameters is specified as follows: <*section*>.<*parameter*> `=` <*value*>. Sections and parameters of non-installed components are not displayed by default. **Arguments** • <*section*>—name of the configuration file section parameters of which are to be displayed. The argument is optional. If not specified, parameters of all configuration file sections are displayed. • <*parameter*>—name of the displayed parameter. Optional argument. If not specified, all parameters of the section are displayed. Otherwise, only this parameter is displayed. If a parameter is specified without the section name, all parameters with this name from all of the configuration file sections are displayed. **Options** `--Uncut`—display all configuration parameters, and not only those used with the currently installed set of components. If the option is not specified, only parameters used by the installed components are displayed. `--Changed`—display only those parameters whose values differ from the default ones. `--Ini`—display parameter values in the `.ini` file format: at first, the section name is specified in square brackets, then the section parameters listed as <*parameter*> `=` <*value*> pairs (one pair per line). `--Value`—output only the value of the specified parameter. The <*parameter*> argument is mandatory in this case.

Command	Description
`reload`	**Purpose**: Reload the configuration of Dr.Web Security Space. For that purpose, the Dr.Web ConfigD configuration management daemon performs the following actions: • rereads the configuration and notifies all Dr.Web Security Space components about its changes; • reopens the Dr.Web Security Space log; • starts the components that use virus databases (including the scanning engine); • attempts to start those components that were shut down abnormally. **Arguments**: None. **Options**: None

3.4. Commands to Manage Detected Threats and Quarantine

The following commands for managing threats and quarantine are available:

Command	Description
`threats [`*<action>* *<object>*`]`	**Purpose**: Apply the specified action to earlier detected threats according to their identifiers. A type of the action is specified by the command option. If the action is not specified, displays information about detected but not neutralized threats. The information about threats is displayed according the format, specified using the non-mandatory `--Format` option. If the `--Format` option is not specified, the following information is displayed for each threat: • an identifier assigned to the threat (its ordinal number); • the full path to the infected file; • information about the threat (its name and type according to the classification of the Doctor Web company); • information about the file: its size, owner, time of last modification; • history of operations applied to an infected file: detection, applied actions and so on. **Arguments**: None. **Options** `--Format "`*<format string>*`"`—output information about threats in the specified format. The description of the format string is below. If this option is specified together with any action option, it is ignored. `-f [--Follow]`—wait for new messages about new threats and display them once they are received (CTRL+C interrupts the waiting).

Command	Description
	If this option is specified together with any action option, it is ignored. `--Directory` <*list of directories*>—output only threats detected in files in directories from <*list of directories*>. If this option is specified together with any option provided below, it is ignored. `--Cure` <*threat list*>—attempt to cure the listed threats (threat identifiers are comma-separated); `--Quarantine` <*threat list*>—quarantine the listed threats (threat identifiers are comma-separated); `--Delete` <*threat list*>—delete the listed threats (threat identifiers are comma-separated); `--Ignore` <*threat list*>—ignore the listed threats (threat identifiers are comma-separated). If you need to apply the action to all detected threats, specify `All` instead of <*threat list*>. For example, the command: `$ drweb-ctl threats --Quarantine All` quarantines all detected malicious objects

`quarantine`
`[`<*action*> <*object*>`]`
	Purpose: Apply an action to the specified object in quarantine.
	If the action is not specified, information about quarantined objects and their identifiers together with brief information about original files put in quarantine is displayed. Information about isolated objects is output according to a format specified with the optional `--Format` parameter. If the `--Format` parameter is not specified, the following information is output for every isolated (quarantined) object:
	• an identifier assigned to a quarantined object;
	• the original path to the file that was moved to quarantine;
	• the date of putting the file in quarantine;
	• information about the file: its size, owner, time of last modification;
	• information about the threat (name of the threat, threat type according to the classification used by the Doctor Web company).
	Arguments: None.
	Options
	`-a [--Autonomous]`—run a standalone instance of the Dr.Web File Checker file scanning component to perform the specified quarantine action and shut down the component upon completion.
	This option can be used together with any options mentioned below.
	`--Format "`<*format string*>`"`—display information about quarantined objects in the specified format. The description of format string is below.
	If this option is specified together with any action option, it is ignored.

Command	Description
	`-f [--Follow]`—wait for new messages about new threats and display them once they are received (CTRL+C interrupts the waiting). If this option is specified together with any action option, it is ignored. `--Discovery [`<*list of directories*>`,]` searches for quarantine directories in the specified list of directories and add them to the consolidated quarantine upon detecting a threat. If the <*list of directories*> is not specified, search for quarantine directories in the common locations of the file system (volume mounting points and user home directories). This option can be specified not only with the `-a` (`--Autonomous`) option (see above), but also with any options/actions listed below. Moreover, if the `quarantine` command is run in standalone instance mode, that is, with the `-a` (`--Autonomous`) option but without the `--Discovery` option, then it has the same effect as calling: `quarantine --Autonomous --Discovery` `--Delete` <*object*>—delete the specified quarantined object. ⚠ Quarantined objects are deleted permanently—this action is irreversible. `--Cure` <*object*>—attempt to cure the specified object in the quarantine. ! Even if the object was successfully cured, it will remain in quarantine. To restore the cured object from quarantine, use the `--Restore` option. `--Restore` <*object*>—restore the specified object from the quarantine to its original location. ! This command may require to run `drweb-ctl` with superuser (the *root* user) privileges. You can restore the file from quarantine even if it is infected. `--TargetPath` <*path*>—restore an object from quarantine to the specified location: either as a file with the specified name (if <*path*> is a path to a file), or to the specified directory (if <*path*> is a path to a directory). A path can be absolute or relative (with regard to the current directory). ⚠ This option can only be used in combination with the `--Restore` option.

Command	Description
	As an <*object*>, specify the object identifier in quarantine. To apply the action to all quarantined objects, specify `All` instead of <*object*>. For example, the command `$ drweb-ctl quarantine --Restore All --TargetPath test` restores all quarantined objects and puts them in the `test` subdirectory located in the current directory from which the `drweb-ctl` command was run. ⚠ If the `--Restore All` variant is indicated together with the additional option `--TargetPath`, this option must set a path to a directory, not to a file.

Formatted output for threats and quarantine commands

The output format is defined using the format string specified as the optional argument `--Format`. The format string must be put in quotes. The format string can include common symbols (displayed “as is”), as well as special markers which will be replaced with corresponding information at the output. The following markers are available:

1. Common for `threats` and `quarantine` commands:

Marker	Description
`%{n}`	New line
`%{t}`	Tabulation
`%{threat_name}`	The name of the detected threat according to the classification of the Doctor Web company
`%{threat_type}`	Threat type (“known virus” and so on) according to Doctor Web classification
`%{size}`	Original file size
`%{origin}`	The full name of the original file with path
`%{path}`	Synonym of `%{origin}`
`%{ctime}`	Modification date/time of the original file in "*%Y-%b-%d %H:%M:%S*" format (for example, `"2018-Jul-20 15:58:01"`)
`%{timestamp}`	Similar to `%{ctime}`, but in the *UNIX timestamp* format
`%{owner}`	The original file owner

Marker	Description
`%{rowner}`	The remote owner of the original file (if not applicable or value is unknown it is replaced with `?`)

2. Specific for `threats` command:

Marker	Description
`%{hid}`	The identifier of the threat record in the history of events associated with the threat
`%{tid}`	Threat identifier
`%{htime}`	Date/time of the event related to the threat
`%{app}`	The identifier of the Dr.Web Security Space component which processed a threat
`%{event}`	The latest event related to a threat: • `FOUND`—threat was detected; • `CURE`—threat was cured; • `QUARANTINE`—file with a threat was quarantined; • `DELETE`—file with a threat was deleted; • `IGNORE`—threat was ignored; • `RECAPTURED`—threat was detected by another component
`%{err}`	Error message text (if no error has occurred, the text is replaced with an empty string)

3. Specific for `quarantine` command:

Marker	Description
`%{qid}`	The identifier of the quarantined object
`%{qtime}`	Date/time of moving the object to quarantine
`%{curetime}`	Date/time of curing attempt of the quarantined object (if not applicable or the value is unknown, it is replaced with `?`)
`%{cureres}`	The result of the quarantined object curing attempt: • `cured`—threat was cured; • `not cured`—threat was not cured or no curing attempts were made

Example

```
$ drweb-ctl quarantine --Format "{%{n} %{origin}: %{threat_name} - %{qtime}%
{n}}"
```

This command displays quarantine contents as records of the following type:

```
{
 <path to file>: <threat name> - <date of putting in quarantine>
}
...
```

3.5. Information Commands

The following information commands are available:

Command	Description
`appinfo`	**Purpose**: Output information about active Dr.Web Security Space components. The following information is output for each running component: • internally used name; • GNU/Linux process identifier (PID); • state (running, stopped and so on); • error code, if the component has been terminated owing to an error; • additional information (optional); For the configuration daemon (`drweb-configd`), the following is output as additional information: • the list of installed components—*Installed*; • the list of components which must be run by the configuration daemon—*Should run*. **Arguments**: None. **Options** `-f [--Follow]`—wait for new messages on module status change and display them once such a message is received (CTRL+C interrupts waiting)
`baseinfo`	**Purpose**: Display the information on the current version of the scan engine and status of virus databases. The following information is displayed: • version of the scan engine; • release date and time of the virus databases being used; • the number of available threat records; • the time of the last successful update of the virus databases and of the scan engine; • the time of the next scheduled automatic update. **Arguments**: None.

Command	Description
	Options `-l [--List]`—display the full list of loaded files of virus databases and a number of threat records in each file
`certificate`	**Purpose**: Display contents of the trusted Dr.Web certificate used by Dr.Web Security Space to scan protected connections if this option is enabled in settings. To save the certificate in the <*cert_name*>`.pem` file, use the command: `$ drweb-ctl certificate >` <*cert_name*>`.pem` **Arguments**: None. **Options**: None
`events`	**Purpose**: Display Dr.Web Security Space events. In addition, this command allows you to manage the events (mark them as read or delete them). **Arguments**: None. **Options** `--Report` <*type*>—type of the event report. Allowed values: • `BRIEF`—brief report; • `DEBUG`—detailed report; • `JSON`—serialized report in the JSON format. `-f [--Follow]`—wait for new events and display them upon their occurrence (CTRL+C interrupts waiting). `-s [--Since]` <*date, time*>—show the events that occurred before the specified timestamp (<*date, time*> is specified as `"YYYY-MM-DD hh:mm:ss"`). `-u [--Until]` <*date, time*>—show the events that occurred no later than the specified timestamp (<*date, time*> is specified as `"YYYY-MM-DD hh:mm:ss"`). `-t [--Types]` <*type list*>—show the events of the specified types only (types are comma-separated). The following event types are available: • `Mail`—threat detection in an email message; • `UnexpectedAppTermination`—unexpected component shutdown. To view all types of events, use `All`. `--ShowSeen`—display already read events as well; `--Show` <*list of events*>—display the listed events (event identifiers are comma-separated);

Command	Description
	`--Delete` <*list of events*>—remove the listed events (event identifiers are comma-separated); `--MarkAsSeen` <*list of events*>—mark the listed events as read (event identifiers are comma-separated). If you want to mark as “read” or delete all events, specify `All` instead of <*events list*>. For example, the command `$ drweb-ctl events --MarkAsSeen All` will mark all existing events as “read”
`report` <*type*>	**Purpose**: Create a report on Dr.Web Security Space events in the HTML format (the page body is output to the specified file). **Arguments** <*type*>—event type that required reporting (indicate one type). See allowed values in the `--Types` option description of the `events` command above. A mandatory argument. **Options** `-o [--Output]` <*path to file*>—save the report to the specified file. The option is mandatory. `-s [--Since]` <*date, time*>—report events that occurred no earlier than the specified timestamp (<*date, time*> is specified as `"YYYY-MM-DD hh:mm:ss"`). `-u [--Until]` <*date, time*>—report events that occurred no later than the specified timestamp (<*date, time*> is specified as `"YYYY-MM-DD hh:mm:ss"`). `--TemplateDir` <*path to directory*>—path to the directory that contains HTML report templates. Options `-s`, `-u`, and `--TemplateDir` are not mandatory. For example, the command: `$ drweb-ctl report Mail -o report.html` generates a report on all existing email message threat detection events, based on the default template, and saves the result in the `report.html` file in the current directory.
`idpass` <*identifier*>	**Purpose:** Display the password generated by the email scanning component Dr.Web MailD for an email message with the indicated identifier and used for the protection of an enclosed archive with threats removed from the email message (i.e. if `RepackPassword` parameter was set in the component settings to `HMAC(`<*secret*>`)`). **Arguments** <*identifier*>—identifier of an email message.

Command	Description
	Options `-s [--Secret]` *<secret>*—secret word used for the generation of the archive password. If a secret word is not indicated when the command is called, the current secret word *<secret>* is used. It is indicated in the Dr.Web MailD settings. If the `RepackPassword` parameter is not available or set to a value different from `HMAC(`*<secret>*`)`, the command will return an error. This command requires `drweb-ctl` to be started with superuser (the *root* user) privileges. If necessary, use the `su` or `sudo` commands.
`license`	**Purpose**: Display the information about the currently active license, get a demo-version license, or get the key file for a license that has already been registered (for example, that has been registered on the company website). If no options are specified, then the following information is output (if you are using a license for the standalone mode): • a license number, • date and time when the license expires. If you are using a license provided to you by a centralized protection server (for the use of the product in the centralized protection mode or mobile mode), the corresponding message is output. **Arguments**: None. **Options** `--GetDemo`—request a demo key that is valid for one month and receive this key, if the conditions for the provision of a demo period have not been violated. `--GetRegistered` *<serial number>*—get a license key file for the specified serial number, if the conditions for the provision of a new key file have not been breached (for example, breached by using the product not in centralized protection mode, when the license is managed by a centralized protection server). `--NetworkTimeout` *<time interval>*—timeout in milliseconds for network operations during the use of the `license` command. This parameter is used to continue activation when the connection is temporarily lost. If the connection is re-established before the timeout expires, the activation will be resumed. If `0` is specified, then there is no timeout. Default value: `0`.

Command	Description
	`--Proxy http://`<*username*>`:`<*password*>`@`<*server address*>`:`<*port*>—get a license key via the proxy server (used only with one of the previously mentioned options — `--GetDemo` or `--GetRegistered`). *If the serial number is not the one provided for a demo period, you must first register this number at the company website.* For further information about licensing Dr.Web products, refer to the Licensing section. ! To register a serial number or to get a demo period, an internet connection is required.
`log`	**Purpose**: Display the latest log records of Dr.Web Security Space in the console (the *stdout* stream, similar to the `tail` command). **Arguments**: None. **Options** `-s [--Size]` <*number*>—number of the latest log records to be displayed on the screen. `-c [--Components]` <*components list*>—list of component identifiers whose records are displayed. Identifiers are defined with comma separation. If no argument is defined, all available records logged by all components are displayed. Actual identifiers of the installed components (e.g. internal component names displayed in the log) can be displayed with the `appinfo` command (see above). `-f [--Follow]`—wait for new log records and display them once they are received (CTRL+C interrupts waiting)
`stat`	**Purpose:** Display statistics about the operation of components that process files or about the operation of the network data scanning agent Dr.Web Network Checker (press CTRL+C or Q to interrupt displaying the statistics). The statistics output includes: • a name of the component that initiated file scanning; • component PID; • an average number of files processed per second during the last minute, 5 minutes, 15 minutes; • a percentage of using the cache of the scanned files; • an average number of scan errors per second. For the distributed scanning agent, the following information is displayed: • a list of local clients that initiated scanning; • a list of remote hosts that received files for scanning; • a list of remote hosts that sent files for scanning.

Command	Description
	For local clients of the distributed scanning agent, their PID and name are specified; for remote clients—an address and port of the host. For both clients—local and remote—the following information is displayed: • an average number of files scanned per second; • an average number of sent and received bytes per second; • an average number of errors per second. **Arguments**: None. **Options** `-n [--netcheck]`—display statistics on operation of the network data scanning agent

9.2.2. Usage Examples

This section contains examples of using the Dr.Web Ctl utility (`drweb-ctl`):

- Scanning Objects:
  - Simple Scanning Commands
  - Scanning of Files Selected by Criteria
  - Scanning of Additional Objects
- Configuration Management
- Threat Management
- Example of Operation in Autonomous Instance Mode
- Updating Offline

1. Scanning Objects

1.1. Simple Scanning Commands

1. Perform scanning of the `/home` directory with default parameters:

```
$ drweb-ctl scan /home
```

2. Scan paths listed in the `daily_scan` file (one path per line):

```
$ drweb-ctl scan --stdin < daily_scan
```

3. Perform scanning of the boot record on the *sda* drive:

```
$ drweb-ctl bootscan /dev/sda
```

4. Perform scanning of the running processes:

```
$ drweb-ctl procscan
```

1.2. Scanning of Files Selected by Criteria

Examples for file selection for scanning are listed below and use the result of the `find` utility operation. The obtained list of files is sent to the `drweb-ctl scan` command with the `--stdin` or `--stdin0` parameter.

1. Scan listed files returned by the `find` utility and separated with the NUL (`\0`) character:

```
$ find -print0 | drweb-ctl scan --stdin0
```

2. Scan all files in all directories, starting from the root directory, on one partition of the file system:

```
$ find / -xdev -type f | drweb-ctl scan --stdin
```

3. Scan all files in all directories, starting from the root directory, with the exception of the `/var/log/messages` and `/var/log/syslog` files:

```
$ find / -type f ! -path /var/log/messages ! -path /var/log/syslog |
drweb-ctl scan --stdin
```

4. Scan all files of the *root* user in all directories, starting from the root directory:

```
$ find / -type f -user root | drweb-ctl scan --stdin
```

5. Scan all files of the *root* and *admin* users in all directories, starting from the root directory:

```
$ find / -type f \( -user root -o -user admin \) | drweb-ctl scan --stdin
```

6. Scan all files of the users with UID within the range of 1000–1005 in all directories, starting from the root directory:

```
$ find / -type f -uid +999 -uid -1006 | drweb-ctl scan --stdin
```

7. Scan files in all directories, starting from the root directory, with a nesting level of no more than five:

```
$ find / -maxdepth 5 -type f | drweb-ctl scan --stdin
```

8. Scan files in a root directory while ignoring files in subdirectories:

```
$ find / -maxdepth 1 -type f | drweb-ctl scan --stdin
```

9. Scan files in all directories, starting from the root directory, while following all symbolic links:

```
$ find -L / -type f | drweb-ctl scan --stdin
```

10. Scan files in all directories, starting from the root directory, without following symbolic links:

```
$ find -P / -type f | drweb-ctl scan --stdin
```

11. Scan files created no later than May 1, 2017 in all directories, starting from the root directory:

```
$ find / -type f -newermt 2017-05-01 | drweb-ctl scan --stdin
```

1.3. Scanning of Additional Objects

1. Scanning of objects located in the `/tmp` directory on the remote host *192.168.0.1* by connecting to it via SSH as the user *user* with the password *passw*:

```
$ drweb-ctl remotescan 192.168.0.1 /tmp --Login user --Password passw
```

2. Scanning of a mail message saved in the file `email.eml`, using the default set of rules:

```
$ drweb-ctl checkmail email.eml
```

2. Configuration Management

1. Display information about the running components of Dr.Web Security Space:

```
$ drweb-ctl appinfo
```

2. Display all parameters of the `[Root]` section:

```
$ drweb-ctl cfshow Root
```

3. Set `No` as the value of the `Start` parameter in the `[LinuxSpider]` section of the active configuration (this will disable SpIDer Guard):

```
# drweb-ctl cfset LinuxSpider.Start No
```

   Note that superuser privileges are required to perform this action. To elevate the privileges, you can use the `sudo` command, as shown in the following example:

```
$ sudo drweb-ctl cfset LinuxSpider.Start No
```

4. Force update of anti-virus components of Dr.Web Security Space:

```
$ drweb-ctl update
```

5. Restart the component configuration of Dr.Web Security Space:

```
# drweb-ctl reload
```

Note that superuser privileges are required to perform this action. To elevate the privileges, you can use the `sudo` command, as shown in the following example:

```
$ sudo drweb-ctl reload
```

6. Connect Dr.Web Security Space to a centralized protection server operating on host *192.168.0.1* if a server certificate is stored in the `/home/user/cscert.pem` file:

   ```
   $ drweb-ctl esconnect 192.168.0.1 --Certificate /home/user/cscert.pem
   ```

7. Connect Dr.Web Security Space to the centralized protection server using the `install.cfg` configuration file:

   ```
   $ drweb-ctl esconnect --cfg <path to install.cfg>
   ```

8. Disconnect Dr.Web Security Space from the centralized protection server:

   ```
   # drweb-ctl esdisconnect
   ```

   Note that superuser privileges are required to perform this action. To elevate the privileges, you can use the `sudo` command, as shown in the following example:

   ```
   $ sudo drweb-ctl esdisconnect
   ```

9. View the last log records made by the `drweb-update` and `drweb-configd` components in the Dr.Web Security Space log:

   ```
   # drweb-ctl log -c Update,ConfigD
   ```

3. Threat Management

1. Display information on detected threats:

   ```
   $ drweb-ctl threats
   ```

2. Quarantine all files containing non-neutralized threats:

   ```
   $ drweb-ctl threats --Quarantine All
   ```

3. Display the list of quarantined files:

   ```
   $ drweb-ctl quarantine
   ```

4. Restore all quarantined files:

   ```
   $ drweb-ctl quarantine --Restore All
   ```

5. Generate a password for a protected archive in the mail message with the identifier `12345`, under condition that, for this email message, the *HMAC* method of password generation has been used, and up-to-date secret word is indicated in the settings of Dr.Web MailD:

   ```
   # drweb-ctl idpass 12345
   ```

4. Example of Operation in Autonomous Instance Mode

Scan files and process quarantine in autonomous instance mode:

```
$ drweb-ctl scan /home/user -a --OnKnownVirus=QUARANTINE
$ drweb-ctl quarantine -a --Delete All
```

The first command will scan files in the `/home/user` directory in autonomous instance mode. Files containing known threats will be quarantined. The second command will process quarantine content (in autonomous instance mode as well) and remove all quarantined objects.

5. Updating Offline

In highly secure environments where internet connection is blocked or limited, it is possible to update virus bases offline. You need to download updates to a computer connected to the internet, copy them to a USB drive or local network share and then install them to another computer (which is not connected to the internet).

The update procedure must run in the command line.

To get updates

1. Run the following command on a computer connected to the internet:

   ```
   $ drweb-ctl update --Path <path to the directory to store updates>
   ```

2. Copy the downloaded updates to a USB drive or a local network share.
3. Mount the local network share or removable drive on the computer to be updated. If the updates are from the USB drive, run the following commands:

   ```
   # mkdir /mnt/usb
   # mount <path to the device> /mnt/usb
   ```

4. Apply the updates with the following command:

   ```
   $ drweb-ctl update --From /mnt/usb
   ```

9.2.3. Configuration Parameters

The Dr.Web Ctl command-line management tool does not have its own parameter section in the unified configuration file of Dr.Web Security Space. The tool uses the parameters specified in the `[Root]` section.

9.3. SpIDer Guard

> This component is included only in the distributions designed for the OSes of the GNU/Linux family.

The SpIDer Guard file system monitor is designed for monitoring file activity on GNU/Linux file system volumes. The component operates as a resident monitor and controls main file system events related to file modification (creating, opening, closing). When such an event is intercepted, the monitor checks whether the file was modified and, if so, the module generates a task for the Dr.Web File Checker file scanning component to scan the modified file with Dr.Web Scanning Engine.

Moreover, the SpIDer Guard file system monitor detects attempts to run executable files. If a program in an executable file is considered malicious during scanning, all processes started from this file will be forcibly terminated.

9.3.1. Operating Principles

In this section:

- General Information.
- Defining Areas of the File System to be Monitored.
- Enhanced File Monitoring Mode.

General Information

The SpIDer Guard file system monitor operates in user space (*user mode*) using the fanotify system mechanism or a custom Linux *loadable kernel module* (*LKM*) developed by Doctor Web. It is recommended that you use the *automatic mode* (`Auto`), which will allow the component to determine and use the best operation mode at startup, because not all Linux kernel versions support fanotify used by the monitor. If the component cannot support the specified integration mode, the components exits after startup. If the auto mode is selected, the component attempts to use the fanotify mode and then the LKM mode. If none of these modes can be used, the component shuts down.

> An already compiled kernel module is distributed with SpIDer Guard for some operating systems. If a kernel module is not compiled for the operating system that uses SpIDer Guard, use module source code provided by Doctor Web to build and install the kernel module manually (for instructions, refer to the Appendix F. Building Kernel Module for SpIDer Guard section).

Once new or modified files are detected, the monitor sends a task to the Dr.Web File Checker component to scan these files. The component then initiates their scanning by Dr.Web Scanning Engine. The operation scheme is shown in the picture below. When operating via the fanotify system mechanism, the monitor can block access to files (of all types of files or only executable files—PE, ELF, scripts with the `#!` preamble) that are not scanned yet until they are scanned (see below).

> SpIDer Guard automatically detects mounting and unmounting new file system volumes (for example, on USB flash drives, CD/DVD, RAID arrays, and so on) and adjusts the list of monitored objects, if necessary.

Defining Areas of the File System to Be Monitored

To optimize file system scanning, SpIDer Guard controls requests only to those files that are in the file system scopes specified in the configuration. Each scope is defined as a path to some directory of the file system tree and is called a *protected space*. A set of all protected spaces forms a single *monitoring scope* controlled by the monitor. Besides the monitoring scope, you can specify a set of file system directories to be excluded from monitoring in the component settings (*exclusion scope*). If you did not specify any protected space in the component settings, the monitoring scope covers the entire file system tree. Thus, only those files are monitored which paths are covered by the monitoring scope and are not covered by the exclusion scope.

Specifying exclusion can be useful when, for example, some files are frequently modified, which results in constant repeated scanning of these files thus increasing the system load. If it is known with certainty that frequent modification of files in a directory is not caused by a malicious program but is due to operation of a trusted program, you can add the path of this directory or files modified in it to the list of exclusions. In this case, the SpIDer Guard file system monitor stops responding to modification of these objects, even if they are covered by the monitoring scope. Moreover, you can add the program processing the files to the list of trusted programs (the `ExcludedProc` configuration parameter). In this case, file operations of this program will not result in scanning even if they are covered by the monitoring scope. Similarly, if necessary, you can disable monitoring and scanning of files from other file systems that are mounted on the local file system (for example, external file server directories mounted via CIFS). To specify file systems which files should not be scanned, use the `ExcludedFilesystem` parameter.

Protected spaces, as parts of the monitoring scope with scan parameters specified for them, are set in the component settings as named sections, which names contain a unique identifier assigned to the protected space. Each section describing a space contains the `Path` parameter that defines a path in the file system storing directories of this protected space (i.e. the file system tree fragment that is monitored within this space) and the `ExcludedPath` parameter that defines a local (with respect to `Path`) exclusion scope inside the protected space. Note that the `ExcludedPath` parameter can contain standard file masks (i.e. `*` and `?` characters). Besides local exclusion scopes, you can also specify a global exclusion scope by using the `ExcludedPath` parameter specified outside the sections describing protected spaces. All directories covered by this scope, including the directories of protected spaces, are excluded from monitoring. Only global and individual exclusion scopes can be applied to each protected space: if a space is nested in another, the exclusion settings specified for the enclosing space are not applied to the nested space. In addition, each protected scope settings have the `Enable` logical parameter which determines whether the files covered by this space are monitored. If this parameter is set to `No`, the contents of this space is not monitored. Moreover, the protected space is not monitored if the `Path` parameter has an empty value.

> If all protected spaces specified in the monitor settings are not monitored or their paths are not specified, SpIDer Guard is running idle because none of the files of the file system tree are monitored. If you want to monitor the file system as a single protected space, remove all named protected space sections from the settings.

Consider an example of configuring monitor and exclusion scopes. Suppose the following parameters are set in the component settings:

```
[LinuxSpider]
ExcludedPath = /directory1/tmp
...

[LinuxSpider.Space.space1]
Path = /directory1
ExcludedPath = "*.tmp"
...

[LinuxSpider.Space.space2]
Path = /directory1/directory2
...

[LinuxSpider.Space.space3]
Path = /directory3
Enable = No
...
```

This means that the files in the `/directory1` directory and its subdirectories, except the `/directory1/tmp` directory, are monitored. Moreover, the files which full names match the `/directory1/*.tmp` mask are not monitored (except the `/directory1/directory2` nested scope to which this mask is not applied despite the fact that the scope is nested in the *space1* protected space). Files in the `/directory3` directory are not monitored.

Enhanced File Monitoring Mode

SpIDer Guard can use three monitoring modes:

- *Regular* (set by default)—monitors file access operations (creation, opening, closing, and running). Scanning of a file, access to which has been allowed, is requested. If a threat is detected upon the scan, the action to neutralize the threat can be applied to the file. Applications are allowed to access the file until the file scanning is finished.

- *Enhanced control of executable files*—monitor files considered non-executable as in regular mode. SpIDer Guard blocks access to an executable file until its scanning is finished.

> Executable files are binary files of PE and ELF formats, as well as text script files containing the "`#!`" preamble.

- *"Paranoid" mode*—SpIDer Guard blocks access to any file until its scanning is finished.

Dr.Web File Checker stores file scan results in specialized cache for a certain time, so reaccessing the same file does not lead to rescanning it if there is information in the cache; the result extracted from the cache is therefore displayed as the scan result. Despite this, the use of the "paranoid" monitoring mode leads to a significant slowdown in accessing files.

To configure the monitoring mode, change the `BlockBeforeScan` parameter value in the component settings.

9.3.2. Command-Line Arguments

To start the SpIDer Guard component from the command line, run the following command:

```
$ /opt/drweb.com/bin/drweb-spider [<parameters>]
```

SpIDer Guard accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-spider --help
```

This command outputs short help information about the SpIDer Guard component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon at the startup of the operating system. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

> To start the Dr.Web Ctl tool, run the `drweb-ctl` command.
>
> To get documentation for this component from the command line, run the `man 1 drweb-spider` command.

9.3.3. Configuration Parameters

The component uses configuration parameters specified in the `[LinuxSpider]` section of the unified configuration file of Dr.Web Security Space.

- Component Parameters.
- Customizing Protected Space Individual Monitoring Settings.

Component Parameters

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-spider`
`Start` *{boolean}*	The component is started by the Dr.Web ConfigD configuration management daemon. Setting this parameter to `Yes` instructs the configuration management daemon to start the component immediately, and setting this parameter to `No`—to shut down the component immediately. Default value: Depends on the Dr.Web product as part of which the component is supplied.
`Mode` *{LKM | FANOTIFY | AUTO}*	SpIDer Guard operation mode.

Parameter	Description
	Allowed values: • `LKM`—use the Dr.Web LKM module installed in the operating system kernel (*LKM — Loadable Kernel Module*); • `FANOTIFY`—use the *fanotify* monitoring interface; • `AUTO`—select an optimal operation mode automatically. ⚠ This parameter value should be changed with extreme caution because Linux kernels support both operation modes to a different degree. It is strongly recommended that you set this parameter value to `AUTO`, as in this case the best mode will be selected for integration with the file system manager at startup. At that, the component will attempt to enable the `FANOTIFY` mode and, on failure,—`LKM`. If none of the modes can be set, the component shuts down. If necessary, you can build the Dr.Web LKM module from source code and install this module by following the instructions in the Appendix F. Building Kernel Module for SpIDer Guard section. Default value: `AUTO`
`DebugAccess` *{boolean}*	Log or do not log detailed information on file access attempts at the debug level (with `LogLevel = DEBUG`). Default value: `No`

`ExcludedProc`
{path to file or path list}
	List of processes which file activity is not monitored. If a file operation was initiated by one of the processes specified in the parameter value, the modified or created file will not be scanned.
	Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all its values are combined into one list). You can use file masks (containing characters `?` and `*` as well as character classes `[ ]`, `[! ]` and `[^ ]`).
	Example: Add the `wget` and `curl` processes to the list.
	• Adding the values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset LinuxSpider.ExcludedProc -a /usr/bin/wget`
	`# drweb-ctl cfset LinuxSpider.ExcludedProc -a /usr/bin/curl`

Parameter	Description

	• Adding values to the configuration file.
	○ Two values per line:
	`[LinuxSpider]`
	`ExcludedProc = "/usr/bin/wget", "/usr/bin/curl"`
	○ Two lines (one value per line):
	`[LinuxSpider]`
	`ExcludedProc = /usr/bin/wget`
	`ExcludedProc = /usr/bin/curl`
	To apply the changes, reload the Dr.Web Security Space configuration using the command:
	`# drweb-ctl reload`
	Default value: *(not specified)*

`ExcludedFilesystem`
{file system name}
	File system accessing the files of which will not be monitored.
	This option is available only in `FANOTIFY` mode.
	Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add the `cifs` and `nfs` file systems to the list.
	• Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset`
	`LinuxSpider.ExcludedFilesystem -a cifs`
	`# drweb-ctl cfset`
	`LinuxSpider.ExcludedFilesystem -a nfs`
	• Adding values to the configuration file.
	○ Two values per line:
	`[LinuxSpider]`
	`ExcludedFilesystem = "cifs", "nfs"`
	○ Two lines (one value per line):
	`[LinuxSpider]`
	`ExcludedFilesystem = cifs`
	`ExcludedFilesystem = nfs`
	To apply the changes, reload the Dr.Web Security Space configuration using the command:

Parameter	Description
	`# drweb-ctl reload` Default value: `cifs`
`BlockBeforeScan` *{Off | Executables | All}*	Block files while being accessed until they are scanned by the monitor (an enhanced or "paranoid" monitoring mode). Allowed values: • `Off`—do not block access to files even if they were not scanned. • `Executables`—block access to executable files (PE and ELF files and scripts containing the `#!` preamble) not scanned by the monitor. • `All`—block access to all files not scanned by the monitor. Files are blocked only in `FANOTIFY` mode. Default value: `Off`

`[*] ExcludedPath`
{path to file or directory}
	Path to an object (file or directory) to be excluded from file monitoring. Either an individual file or an entire directory can be specified. If a directory is specified, all files and subdirectories (including nested ones) will be skipped. You can use file masks (containing characters `?` and `*` as well as character classes `[ ]`, `[! ]` and `[^ ]`).
	Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add the `/etc/file1` file and the `/usr/bin` directory to the list.
	• Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset LinuxSpider.ExcludedPath -a /etc/file1`
	`# drweb-ctl cfset LinuxSpider.ExcludedPath -a /usr/bin`
	• Adding values to the configuration file.
	○ Two values per line:
	`[LinuxSpider]`
	`ExcludedPath = "/etc/file1", "/usr/bin"`
	○ Two lines (one value per line):
	`[LinuxSpider]`
	`ExcludedPath = /etc/file1`
	`ExcludedPath = /usr/bin`

Parameter	Description
	To apply the changes, reload the Dr.Web Security Space configuration using the command: `# drweb-ctl reload` There is no point in providing paths to symbolic links here as only a direct path to a file is analyzed while scanning it. Default value: `/proc, /sys`

`[*] OnKnownVirus`
{action}
	Action to be applied upon detection of a known threat (a virus and so on) in the scanned file.
	Allowed values: `CURE`, `QUARANTINE`, `DELETE`.
	Default value: `CURE`

`[*] OnIncurable`
{action}
	Action to be applied upon detection of an incurable threat.
	Allowed values: `QUARANTINE`, `DELETE`.
	Default value: `QUARANTINE`

`[*] OnSuspicious`
{action}
	Action to be applied upon detection of an unknown threat (or a suspicious object) in the scanned file by using heuristic analysis.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.
	Default value: `QUARANTINE`

`[*] OnAdware`
{action}
	Action to be applied upon detection of adware in the scanned file.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.
	Default value: `QUARANTINE`

`[*] OnDialers`
{action}
	Action to be applied upon detection of a dialer in the scanned file.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.
	Default value: `QUARANTINE`

`[*] OnJokes`
{action}
	Action to be applied upon detection of a joke program in the scanned file.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.
	Default value: `REPORT`

`[*] OnRiskware`
{action}
	Action to be applied upon detection of riskware in the scanned file.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.

Parameter	Description
	Default value: `REPORT`

`[*] OnHacktools`
{action}
	Action to be applied upon detection of a hacktool in the scanned file.
	Allowed values: `REPORT`, `QUARANTINE`, `DELETE`.
	Default value: `REPORT`

`[*] ScanTimeout`
{time interval}
	Timeout for scanning one file.
	Allowed values: from 1 second (`1s`) to 1 hour (`1h`).
	Default value: `30s`

`[*] HeuristicAnalysis`
{On | Off}
	Enable or disable the heuristic analysis for detection of unknown threats. The heuristic analysis provides higher detection reliability but increases the duration of scanning.
	Action applied to threats detected by the heuristic analyzer is specified by the `OnSuspicious` parameter.
	Allowed values:
	• `On`—enable the heuristic analysis while scanning.
	• `Off`—disable the heuristic analysis.
	Default value: `On`

`[*] PackerMaxLevel`
{integer}
	Maximum nesting level for packed objects. A packed object is executable code compressed with special software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects that may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which packed objects inside other packed objects are not scanned.
	The nesting level is not limited. If the value is set to `0`, nested objects are not scanned.
	Default value: `8`

`[*] ArchiveMaxLevel`
{integer}
	Maximum nesting level for archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned.
	The nesting level is not limited. If the value is set to `0`, nested objects are not scanned.
	Default value: `0`

`[*] MailMaxLevel`
{integer}
	Maximum nesting level for files of mailers (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned.

Parameter	Description
	The nesting level is not limited. If the value is set to `0`, nested objects are not scanned. Default value: `0`

`[*] ContainerMaxLevel`
{integer}
	Maximum nesting level while scanning other types of objects containing nested objects (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects will not be scanned.
	The nesting level is not limited. If the value is set to `0`, nested objects are not scanned.
	Default value: `8`

`[*] MaxCompressionRatio`
{integer}
	Maximum compression ratio of scanned objects (a ratio of an uncompressed size to a compressed size). If the ratio of an object exceeds the limit, this object is skipped during the scan.
	The compression ratio must be no less than `2`.
	Default value: `500`

Customizing Protected Space Individual Monitoring Settings

For each protected space of a file system, a separate section containing the path to a monitored file system area and monitoring parameters is specified in the configuration file together with the `[LinuxSpider]` section, which stores all the monitor parameters. Each section must be named as `[LinuxSpider.Space.<`*space name*`>]`, where <*space name*> is a unique identifier of the protected space.

The space individual section must contain the following parameters absent in the `[LinuxSpider]` general section:

Parameter	Description
`Enable` *{boolean}*	Contents of the protected space located at `Path` (see below) must be monitored. To stop monitoring the contents of this protected space, set the parameter to `No`. Default value: `Yes`
`Path` *{path to directory}*	Path to a directory with files to be monitored (including nested directories). By default, this parameter has an empty value; therefore, you must specify a value when adding the protected space to the monitoring scope. Default value: *(not specified)*

> If all protected spaces specified in the monitor settings are not monitored or their paths are not specified, SpIDer Guard is running idle because none of the files of the file system tree are monitored. If you want to monitor the file system as a single protected space, remove all named protected space sections from the settings.

Except for those mentioned above, separate sections of protected spaces can include a list of parameters from the general section of the component settings that are marked with the `[*]` designation in the table above and redefine a parameter specified for the protected space (for example, an action to be applied upon threat detection, the maximum archive nesting level and so on). If the parameter is not specified for the protected space, file monitoring for this space is adjusted with the corresponding parameter values from the `[LinuxSpider]` section.

To add a new section of parameters for the protected space with the <*space name*> tag using the Dr.Web Ctl management tool (started with the `drweb-ctl` command), run the command:

```
# drweb-ctl cfset LinuxSpider.Space -a <space name>
```

Example:

```
# drweb-ctl cfset LinuxSpider.Space -a Space1
# drweb-ctl cfset LinuxSpider.Space.Space1.Path /home/user1
```

The first command adds the `[LinuxSpider.Space.Space1]` section to the configuration file; the second one sets a value of the `Path` parameter for the section by specifying the path to the monitored file system area. Other parameters of this section will be the same as in the `[LinuxSpider]` general section.

9.4. Dr.Web MailD

The Dr.Web MailD component is designed for direct email scanning, detection of malicious contents (not only attachments but also links to malicious or unwanted websites), analyzing messages for signs of spam and checking their compliance with the security criteria set by an email system administrator (scanning of email message body and headers using regular expressions specified by the administrator).

> If the scanning of email messages is highly intense, scanning issues can occur due to depletion of the number of available file descriptors by the Dr.Web Network Checker component. In this case, it is necessary to increase the limit by the number of file descriptors available to Dr.Web Security Space.

9.4.1. Operating Principles

The component can protect email messages by setting up *proxy* that performs scanning of emails transferred via SMTP, POP3 or IMAP4 protocols transparently for the mail server. To set up this scanning method, SpIDer Gate and Dr.Web Firewall for Linux are used. As these components operate only with GNU/Linux, this method is available only for this family of operating systems.

The component uses the processing rules determined in the settings of Dr.Web Firewall for Linux.

For scanning the URLs in email messages, the same databases of web resource categories as in SpIDer Gate component, are used. Dr.Web CloudD component is used to refer to Dr.Web Cloud service (the use of the cloud service is configured in Dr.Web Security Space common settings and can be disabled, if necessary). To check transmitted data, Dr.Web MailD uses the Dr.Web Network Checker component. The latter one initiates scanning via the Dr.Web Scanning Engine scan engine.

Scanning email attachments for malicious code is performed directly by Dr.Web MailD.

For messages analysis on presence of signs of spam, Dr.Web MailD uses the special component Dr.Web Anti-Spam.

> Depending on the distribution, Dr.Web Anti-Spam can be unavailable in Dr.Web Security Space. In this case, email messages will not be scanned for signs of spam.

9.4.2. Command-Line Arguments

To start the Dr.Web MailD component from the command line, run the command:

```
$ /opt/drweb.com/bin/drweb-maild [<parameters>]
```

Dr.Web MailD accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v`

	Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-maild --help
```

This command outputs short help information about the Dr.Web MailD component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon upon receiving requests on scanning of email objects from other components of Dr.Web Security Space. To manage the operation of the component, as well as to scan email messages when necessary, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To scan an arbitrary email message with the Dr.Web MailD component, use the `checkmail` command of the Dr.Web Ctl tool. To do that, save the scanned email message to a drive (for example, in the `.eml` format) and use the command:

```
$ drweb-ctl checkmail <path to .eml file>
```

To get documentation for this component from the command line, run the `man 1 drweb-maild` command.

9.4.3. Configuration Parameters

The component uses configuration parameters specified in the `[MailD]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	**Description**
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log`	Logging method of the component.

Parameter	**Description**
{log type}	Default value: `Auto`
`ExePath` *{path to file}*	Executable path to the component. Default value: `/opt/drweb.com/bin/drweb-maild`
`FixedSocketPath` *{path to file}*	Path to the UNIX socket file of the component fixed instance. If this parameter is specified, the Dr.Web ConfigD configuration management daemon ensures that there is always a running component instance available to clients via this socket. Default value: *(not specified)*
`TemplatesDir` *{path to directory}*	Path to a directory that contains email message templates returned to the user in case of email blocking. Default value: `/var/opt/drweb.com/templates/maild`

`ReportLanguages`
{string}
	Languages used for generation of service messages (for example, messages returned to the sender in case of email blocking). Each language is identified with a two-letter designation (`en`, `ru` and so on).
	Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add the following languages to the list: `ru` and `de`.
	1. Adding values to the configuration file.
	• Two values per line:
	`[MailD]`
	`ReportLanguages = "ru", "de"`
	• Two lines (one value per line):
	`[MailD]`
	`ReportLanguages = ru`
	`ReportLanguages = de`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset MailD.ReportLanguages -a ru`
	`# drweb-ctl cfset MailD.ReportLanguages -a de`
	Default value: `en`

Parameter	Description

`RepackPassword`
{None | Plain(<password>) | HMAC(<secret>)}
	The method for generation of a password for archives with malicious objects placed in messages and sent to recipients. The following methods are allowed:
	• `None`—archives will not be protected with a password (not recommended);
	• `Plain(`*<password>*`)`—all archives will be protected with the same password *<password>*;
	• `HMAC(`*<secret>*`)`—the unique password will be generated for each archive based on the pair (*<secret>*, *<message identifier>*).
	To recover the password that protects the archive using the message identifier and the known secret, use the `drweb-ctl idpass` command.
	⚠ By default, this parameter has the `None` value; it is recommended to change this value in the course of Dr.Web Security Space configuration.
	Default value: `None`

`TemplateContacts` *{string}*	Dr.Web Security Space administrator contacts to be inserted into messages about threats (used in message templates). The contact information is added to a repacked message only if it has an attachment with a password-protected archive with threats or other unwanted objects removed from the initial message. If, according to the current value of the `RepackPassword` parameter (see below), attached archives are not protected with a password, then the contact information is not added to the modified message. Default value: *(not specified)*
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the "`name:`" prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`DnsResolverConfPath` *{path to file}*	Path to the DNS configuration file (DNS resolver). Default value: `/etc/resolv.conf`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down.

Parameter	Description
	The `IdleTimeLimit` parameter value is ignored (the component does not shut down after the time interval expires), if any of the following parameters is defined: `FixedSocketPath`, `MilterSocket`, `SpamdSocket`, `RspamdHttpSocket`, `RspamdSocket`, `SmtpSocket` or `BccSocket`. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`SpoolDir` *{path to directory}*	Directory for temporary storage of scanned email messages. Default value: `/tmp/com.drweb.maild`
`Hostname` *{string}*	Sender's host name (FQDN). It will appear in the HELO/EHLO welcome string received from the SMTP client and as the default value of `srvname` in the `Authentication-Results` title. Default value: *current host name*
`CaPath` *{path to file or directory}*	Path to the directory or file with a list of trusted root certificates. Default value: *path to the system list of trusted certificates*. The path depends on your GNU/Linux distribution. • For Astra Linux, Debian, Linux Mint, SUSE Linux and Ubuntu, this is usually the path `/etc/ssl/certs/`. • For CentOS and Fedora—the path `/etc/pki/tls/certs/ca-bundle.crt`. • For other distributions, the path can be determined by running the `openssl version -d` command. • If this command is unavailable or your OS distribution cannot be identified, the `/etc/ssl/certs/` value is used
`WarnOfUnknownDomain` *{boolean}*	Add a warning to exercise caution to the email message body, because the sender's domain is not on the list of protected domains (refer to the `ProtectedDomains` parameter). The warning message is specified in the `ExternalDomainWarning` parameter. Allowed values: • `On`, `Yes`, `True`—add the warning; • `Off`, `No`, `False`—do not add the warning. Default value: `No`
`ProtectedDomains` *{string}*	List of domains protected with Dr.Web products. If the sender's domain is not on this list, a warning to exercise caution is added to the email message body (refer to the `WarnOfUnknownDomain` and `ExternalDomainWarning` parameters).

Parameter	Description

	Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add domains `drweb.com` and `drweb.ru` to the list.
	1. Adding values to the configuration file.
	• Multiple values per line:
	`[MailD]`
	`ProtectedDomains = "localhost", "drweb.com", "drweb.ru"`
	• One value per line:
	`[MailD]`
	`ProtectedDomains = localhost`
	`ProtectedDomains = drweb.com`
	`ProtectedDomains = drweb.ru`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset MailD.ProtectedDomains -a drweb.com`
	`# drweb-ctl cfset MailD.ReportLanguages -a drweb.ru`
	Default value: `localhost`

`ExternalDomainWarning`
{encoding; warning message}
	Message of the warning to exercise caution to be added to the email message body if the sender's domain is not on the list of protected domains (refer to the `ProtectedDomains` parameter) and the value of the `WarnOfUnknownDomain` parameter is `Yes`.
	The encoding of the email message is specified in the `charset` field of the email message header. The list of encodings complying with RFC2047 is available at https://www.iana.org/assignments/character-sets/character-sets.xhtml. If the encoding cannot be determined, the `default` value is used. For this case, it is recommended to compose the warning message using the Latin alphabet. If multiple values with the same encoding are specified, the text provided in the last of such values will be used for such encoding.
	Example: Add a warning message for the KOI8-R encoding (also known as `csKOI8R`).
	1. Adding values to the configuration file, one value per line:

Parameter	Description

	`[MailD]`
	`ExternalDomainWarning = "{""default"", ""Attention: The message was sent from an external domain. It is not recommended to follow links, open attachments, or provide confidential information.""}"`
	`ExternalDomainWarning = "{""utf-8"", ""Внимание: письмо отправлено с внешнего домена. Не рекомендуется переходить по ссылкам, открывать вложения или предоставлять конфиденциальную информацию.""}"`
	`ExternalDomainWarning = "{""KOI8-R"", ""External domain warning""}"`
	`ExternalDomainWarning = "{""csKOI8R"", ""External domain warning""}"`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset -a MailD.ExternalDomainWarning '{"KOI8-R", "External domain warning"}'`
	`# drweb-ctl cfset -a MailD.ExternalDomainWarning '{"csKOI8R", "External domain warning"}'`
	Default values:
	`{"default", "Attention: The message was sent from an external domain. It is not recommended to follow links, open attachments, or provide confidential information."}`
	`{"utf-8", "Внимание: письмо отправлено с внешнего домена. Не рекомендуется переходить по ссылкам, открывать вложения или предоставлять конфиденциальную информацию."}`

`ScanTimeout` *{time interval}*	Time-out for scanning one email message. Allowed values: from 1 second (`1s`) to 1 hour (`1h`). Default value: `3m`
`HeuristicAnalysis` *{boolean}*	Enable or disable heuristic analysis for detecting unkown threats while performing a message scan initiated by Dr.Web MailD. Heuristic analysis provides higher detection rates, but at the same time increases scanning duration. Allowed values: • `On`, `Yes`, `True`—enable the heuristic analysis; • `Off`, `No`, `False`—disable the heuristic analysis. Default value: `On`

Parameter	Description
`PackerMaxLevel` *{integer}*	Maximum nesting level for packed objects. A packed object is executable code compressed with special software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects that may also include packed objects and so on. The value of this parameter specifies a nesting limit beyond which packed objects inside other packed objects are not scanned. The nesting level is not limited. If the value is set to 0, nested objects are not scanned. Default value: `8`
`ArchiveMaxLevel` *{integer}*	Maximum nesting level for archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies a nesting limit beyond which archives enclosed in other archives are not scanned. The nesting level is not limited. If the value is set to `0`, nested objects are not scanned. Default value: `8`
`MailMaxLevel` *{integer}*	Maximum nesting level for mailer files (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The value of this parameter specifies a nesting limit beyond which objects inside other objects are not scanned. The nesting level is not limited. If the value is set to `0`, nested objects are not scanned. Default value: `8`
`ContainerMaxLevel` *{integer}*	Maximum nesting level for other types of objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies a nesting limit beyond which objects inside other objects are not scanned. The nesting level is not limited. If the value is set to `0`, nested objects are not scanned. Default value: `8`
`MaxCompressionRatio` *{integer}*	Maximum compression ratio of compressed/packed objects (the ratio of an uncompressed size to a compressed size). If the ratio exceeds the limit, this object is skipped during the scanning initiated by Dr.Web MailD. The compression ratio must be no less than `2`. Default value: `500`

Parameter	Description
`MaxSizeToExtract` *{size}*	Maximum size for files enclosed in archives. Files whose size is greater than the value of this parameter are skipped during scanning. There is no size limit for files in archives by default. The value of this parameter is specified as a number with a suffix (`b`, `kb`, `mb`, `gb`). If no suffix is specified, the value is treated as a size in bytes. If the value is set to `0`, files in archives are not scanned. Default value: `None`
`MilterDebugIpc` *{boolean}*	Log or do not log *Milter* messages at the debug level (`LogLevel = Debug`). Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log. Default value: `No`
`MilterTraceContent` *{boolean}*	Log or do not log body of email messages received for scanning via the *Milter* interface at the debug level (`LogLevel = Debug`). Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log. Default value: `No`

`MilterSocket`
{path to file | IP address:port}
	Socket to connect to an MTA as a *Milter* filter of email messages (the MTA connects to this socket while using Dr.Web MailD as the corresponding filter). Usage of a UNIX socket or a network socket is allowed.
	The rules for processing messages passed via *Milter* are specified in the `MilterHook` parameter (see below).
	Default value: *(not specified)*

`MilterHook`
{path to file | Lua function}
	Lua script for processing email messages received via the *Milter* interface or a path to the file containing this script.
	If the file path is invalid, an error is returned while starting the component.
	Default value:
	`local dw = require "drweb"`
	`local dwcfg = require "drweb.config"`
	
	`function milter_hook(ctx)`

```
  -- Reject the message if it is likely spam
  if ctx.message.spam.score >= 100 then
    dw.notice("Spam score: " ..
ctx.message.spam.score)
    return {action = "reject"}
  else
    -- Assign X-Drweb-Spam headers on the basis
of the spam report
    ctx.modifier.add_header_field("X-DrWeb-
SpamScore", ctx.message.spam.score)
    ctx.modifier.add_header_field("X-DrWeb-
SpamState", ctx.message.spam.type)
    ctx.modifier.add_header_field("X-DrWeb-
SpamDetail", ctx.message.spam.reason)
    ctx.modifier.add_header_field("X-DrWeb-
SpamVersion", ctx.message.spam.version)
  end

  -- Scan the message for threats and repack it
if they are present
  for threat, path in
ctx.message.threats{category = {"known_virus",
"virus_modification", "unknown_virus", "adware",
"dialer"}} do
    ctx.modifier.repack()
    dw.notice(threat.name .. " found in " ..
(ctx.message.part_at(path).name or path))
  end

  -- Repack if an unwanted URL has been found
  for url in ctx.message.urls{category =
{"infection_source", "not_recommended",
"owners_notice"}} do
    ctx.modifier.repack()
    dw.notice("URL found: " .. url .. "(" ..
url.categories[1] .. ")")
  end

  -- Add the X-AntiVirus header
  ctx.modifier.add_header_field("X-AntiVirus",
"Checked by Dr.Web [MailD version: " ..
dwcfg.maild.version .. "]")

  -- Accept the message with all scheduled
transformations applied
  return {action = 'accept'}
end
```

Parameter	Description
`SpamdDebugIpc` *{boolean}*	Log or do not log *Spamd* messages at the debug level (`LogLevel = Debug`). Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log.

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td></td><td>Default value: <code>No</code></td></tr>
<tr><td><code>SpamdSocket</code>

<i>{path to file | IP address:port}</i></td><td>Socket to connect to an MTA as a <i>Spamd</i> filter of email messages (the MTA connects to this socket while using Dr.Web MailD as the corresponding filter). Usage of a UNIX socket or a network socket is allowed.

The rules for processing messages passed via <i>Spamd</i> are specified in the <code>SpamdReportHook</code> parameter (see below).

Default value: <i>(not specified)</i></td></tr>
<tr><td><code>SpamdReportHook</code>

<i>{path to file | Lua function}</i></td><td>Lua script that processes email messages received via the <i>Spamd</i> interface or a path to the file containing the script.

If the file path is invalid, an error is returned while starting the component.

Default value:<pre>
local dw = require "drweb"

function spamd_report_hook(ctx)
  local score = 0
  local report = ""

  -- Add 1000 to the score for each threat found in the message
  for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification", "unknown_virus", "adware", "dialer"}} do
      score = score + 1000
      report = report .. "Threat found: " .. threat.name .. "\n"
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
  end

  -- Add 100 to the score for each unwanted URL found in the message
  for url in ctx.message.urls{category = {"infection_source", "not_recommended", "owners_notice"}} do
      score = score + 100
      report = report .. "Url found: " .. url .. "\n"
      dw.notice("URL found: " .. url .. "(" .. url.categories[1] .. ")")
  end

  -- Add the spam score
  score = score + ctx.message.spam.score
  report = report .. "Spam score: " .. ctx.message.spam.score .. "\n"
</pre></td></tr>
</table>

Parameter	Description

	`if ctx.message.spam.score >= 100 then`
	`dw.notice("Spam score: " ..`
	`ctx.message.spam.score)`
	`end`
	
	`-- Return the scan result`
	`return {`
	`score = score,`
	`threshold = 100,`
	`report = report`
	`}`
	`end`

`RspamdDebugIpc` *{boolean}*	Log or do not log *Rspamd* messages at the debug level (`LogLevel = Debug`). Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log. Default value: `No`

`RspamdHttpSocket`
{path to file | IP address:port}
	Socket to connect to an MTA as an *Rspamd* mail filter (the MTA connects to this socket while using Dr.Web MailD as the corresponding filter, by using the HTTP version of the *Rspamd* protocol). Usage of a UNIX socket or a network socket is allowed.
	The rules for processing messages passed through *Rspamd* are specified in the `RspamdHook` parameter (see below).
	Default value: *(not specified)*

`RspamdSocket`
{path to file | IP address:port}
	Socket to connect to an MTA as an *Rspamd* mail filter (the MTA connects to this socket while using Dr.Web MailD as the corresponding filter, by using the *legacy* version of the *Rspamd* protocol). Usage of a UNIX socket or a network socket is allowed.
	Default value: *(not specified)*

`RspamdHook`
{path to file | Lua function}
	Lua script that processes email messages received via the *Rspamd* interface or a path to the file containing the script.
	If the file path is invalid, an error is returned while starting the component.
	Default value:
	`local dw = require "drweb"`
	
	`function rspamd_hook(ctx)`
	`local score = 0`
	`local symbols = {}`
	
	`-- Add 1000 to the score for each threat found`
	`in the message`

```
  for threat, path in
ctx.message.threats{category = {"known_virus",
"virus_modification", "unknown_virus", "adware",
"dialer"}} do
      score = score + 1000
      table.insert(symbols, {name = threat.name,
score = 1000})
      dw.notice(threat.name .. " found in " ..
(ctx.message.part_at(path).name or path))
  end

  -- Add 100 to the score for each unwanted URL
found in the message
  for url in ctx.message.urls{category =
{"infection_source", "not_recommended",
"owners_notice"}} do
      score = score + 100
      table.insert(symbols, {name = "URL " ..
url, score = 100})
      dw.notice("URL found: " .. url .. "(" ..
url.categories[1] .. ")")
  end

  -- Add the spam score
  score = score + ctx.message.spam.score
  table.insert(symbols, {name = "Spam score",
score = ctx.message.spam.score})
  if ctx.message.spam.score >= 100 then
      dw.notice("Spam score: " ..
ctx.message.spam.score)
  end

  -- Return the scan result
  return {
     score = score,
     threshold = 100,
     symbols = symbols
  }
end
```

Parameter	Description
`SpfCheckTimeout` *{time interval}*	Maximum total time for the SPF check. Default value: `20s`
`SpfVoidLimit` *{integer}*	Maximum number of empty answers allowed during the SPF check. Default value: `2`
`SmtpDebugIpc` *{boolean}*	Log or do not log SMTP messages at the debug level (with `LogLevel = DEBUG`) in SMTP mode. Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log.

Parameter	Description
	Default value: `No`
`SmtpTraceContent` *{boolean}*	Log or do not log email content at the debug level (with `LogLevel = DEBUG`) in SMTP mode. Allowed values: • `On`, `Yes`, `True`—log; • `Off`, `No`, `False`—do not log. Default value: `No`
`SmtpRetryInterval` *{time interval}*	Time-out for a repeated attempt of scanning or sending a message in case of an error while operating in SMTP mode. Allowed values: from 1 second (`1s`) to 1 day (`1d`). Default value: `5m`
`SmtpRequireTls` *{Always | IfSupported | Never}*	SMTP policy for using the STARTTLS extension in SMTP mode. Allowed values: • `Always`—always use a protected connection; interrupt the connection if the server does not support its protection. • `IfSupported`—prefer a protected connection if the server supports it; otherwise, send messages via unprotected channels. • `Never`—do not use unprotected connection. Default value: `Always`
`SmtpSslCertificate` *{path to certificate file}*	Path to a certificate file for connecting an MTA to Dr.Web MailD operating as an external email filter in SMTP or BCC mode. Default value: *(not specified)*
`SmtpSslKey` *{path to private key file}*	Path to a private key file for connecting an MTA to Dr.Web MailD operating as an external email filter in SMTP or BCC mode. Default value: *(not specified)*
`SmtpTimeout` *{time interval}*	Maximum time interval (in minutes) during which scanning must be performed if Dr.Web MailD operates in SMTP mode. If the scanning has not been performed during the specified time interval, the action specified in the `SmtpTimeoutAction` parameter will be performed. If a zero value is specified, the scanning is performed immediately after adding the message to the queue or after another failed scanning attempt. If this parameter value is greater than an `SmtpRetryInterval` parameter value, two scanning attempts will be made within the specified time interval. Allowed values: `0` or from 1 minute (`1m`) to 1 week (`1w`). Default value: `1h`

Parameter	Description
`SmtpTimeoutAction` *{Accept|Discard}*	Action to be performed after the time interval specified in the `SmtpTimeout` parameter expires. Allowed values: • `Accept`—accept (allow the MTA to send the message to the recipient); • `Discard`—discard the message without notifying the sender. Default value: `Accept`

`SmtpSocket`
{path to file | IP address:port}
	Socket to connect to an MTA as a mail filter in SMTP mode (the MTA connects to this socket while using Dr.Web MailD as an external filter). Usage of a UNIX socket or a network socket is allowed. The server connecting via this socket uses a certificate whose path is specified in the `SmtpSslCertificate` parameter and a private key whose path is specified in the `SmtpSslKey` parameter.
	Default value: *(not specified)*

`SmtpSenderRelay`
{path to file | IP address:port}
	Socket to connect Dr.Web MailD to an MTA to send messages that passed scanning in SMTP mode (Dr.Web MailD acting as an external filter connects to the MTA via this socket). Usage of a UNIX socket or a network socket is allowed.
	Default value: *(not specified)*

`SmtpHook`
{path to file | Lua function}
	Lua script that processes email messages received for scanning in SMTP mode or a path to the file containing this script.
	If the `UseVxcube=Yes` parameter value is specified in the `[Root]` section of the configuration file, the step of scanning email attachments with Dr.Web vxCube is added to the Lua script by default.
	Default value:
	`local dw = require "drweb"`
	
	`function smtp_hook(ctx)`
	`  -- Reject the message if it is likely spam`
	`  if ctx.message.spam.score >= 100 then`
	`      dw.notice("Spam score: " .. ctx.message.spam.score)`
	`      return {action = "discard"}`
	`  else`
	`      -- Assign X-Drweb-Spam headers on the basis of the spam report`
	`      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)`
	`      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)`
	`      ctx.modifier.add_header_field("X-DrWeb-`

Parameter	Description

	SpamDetail", ctx.message.spam.reason)
	      ctx.modifier.add_header_field("X-DrWeb-SpamVersion", ctx.message.spam.version)
	  end
	
	  -- Scan the message for threats and repack it if they are present
	  threat_categories = {"known_virus", "virus_modification", "unknown_virus", "adware", "dialer"}
	  if ctx.message.has_threat({category = threat_categories}) then
	      for threat, path in ctx.message.threats({category = threat_categories}) do
	        dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
	      end
	      ctx.modifier.repack()
	      return {action = "accept"}
	  end
	
	  -- Repack if an unwanted URL has been found
	  url_categories = {"infection_source", "not_recommended", "owners_notice"}
	  if ctx.message.has_url({category = url_categories}) then
	      for url in ctx.message.urls({category = url_categories}) do
	        dw.notice("URL found: " .. url .. " (" .. url.categories[1] .. ")")
	      end
	      ctx.modifier.repack()
	      return {action = "accept"}
	  end
	
	  -- Accept the message with all scheduled transformations applied
	  return {action = 'accept'}
	end

BccSocket *{path to file | IP address:port}*	Socket to connect to an MTA as a mail filter in BCC mode (the MTA connects to this socket while using Dr.Web MailD as an external filter). Usage of a UNIX socket or a network socket is allowed. The server connecting via this socket uses a certificate whose path is specified in the SmtpSslCertificate parameter and a private key whose path is specified in the SmtpSslKey parameter. Default value: *(not specified)*
BccReporterAddress *{string}*	Email address from which Dr.Web MailD reports will be sent after scanning email attachments in BCC mode. Default value: *(not specified)*

Parameter	Description

`BccReporterPassword`
{None | Plain(<password>)}
	Password for a mailbox using which Dr.Web MailD reports will be sent after scanning email attachments in BCC mode.
	Allowed values:
	• `None`—email is not protected by a password;
	• `Plain(`*<password>*`)`—mailbox is protected with the specified password.
	Default value: `None`

`BccReportRecipientAddress` *{string}*	Email address to which Dr.Web MailD reports will be sent after scanning email attachments in BCC mode. Default value: *(not specified)*
`BccSmtpServer` *{string}*	MTA address for sending email messages in SMTP and BCC modes. Usage of a domain, an IP address or a UNIX socket is allowed. Default value: *(not specified)*
`BccTimeout`	Maximum time interval (in minutes) during which scanning must be performed if Dr.Web MailD operates in BCC mode. If the scanning has not been performed during the specified time interval, the `Discard` action will be performed. If a zero value is specified, the scanning is performed immediately after adding the message to the queue or after another failed scanning attempt. If this parameter value is greater than an `SmtpRetryInterval` parameter value, two scanning attempts will be made within the specified time interval. Allowed values: `0` or from 1 minute (`1m`) to 1 week (`1w`). Default value: `1h`

`VxcubePlatforms`
{platform, … | All}
	List of OS platforms for executing email attachments while using Dr.Web vxCube as an email message scanning tool in external filter mode (SMTP or BCC).
	The values of the list must be comma-separated (each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Allowed values:
	• *<platform>*—the value of the `os_code` field (the OS name with its bitness specified) from the `platforms` API call in Dr.Web vxCube (for details, refer to the User manual for Dr.Web vxCube, the Platform section);
	• `All`—all available platforms.
	Default value: `All`

Parameter	Description

`VxcubeFileFormats`
{format, … | All}
	List of email attachment formats to be sent for analysis while using Dr.Web vxCube as an email message scanning tool in external filter mode (SMTP or BCC).
	The values of the list must be comma-separated (each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Allowed values:
	• <*format*>—the value of the `name` field (format designation) from the `formats` API call in Dr.Web vxCube (for details, refer to the User manual for Dr.Web vxCube, the Format section);
	• `All`—all available formats.
	Default value: `All`

`VxcubeSampleRunTime` *{time interval}*	Time for executing an email attachment sent for analysis to Dr.Web vxCube while using it as an email message scanning tool in external filter mode (SMTP or BCC). Default value: *(not specified)*

9.5. Dr.Web Anti-Spam

The Dr.Web Anti-Spam component is designed to directly scan email messages for signs of spam. This components is used by the Dr.Web MailD mail scanning component. Depending on package, Dr.Web Anti-Spam can be absent in Dr.Web Security Space (in this case, Dr.Web MailD does not perform spam scans).

> ! The component is not supported for ARM64, E2K and IBM POWER (ppc64el) architectures.

9.5.1. Operating Principles

The analysis of messages received from Dr.Web MailD (or any other external application) for signs of spam is performed using the anti-spam library and the Dr.Web Anti-Spam component. The analysis of the messages is performed in standalone mode without requests to external sources of information on spam. This solution provides a high rate of message processing and constant improvement of message analysis owing to the dynamic update of the database of rules for spam classification of messages (the update is performed automatically via Dr.Web Updater).

You can create your own component (an external application) using Dr.Web Anti-Spam for scanning email messages for spam. For this, Dr.Web Anti-Spam provides a special API based on Google Protobuf. To obtain the Dr.Web Anti-Spam API guide and examples of client application code using Dr.Web Anti-Spam, contact the partner relations department of the Doctor Web company (https://partners.drweb.com/).

Dr.Web Security Space is not distributed with the Dr.Web Anti-Spam component on ARM64, E2K and IBM POWER (ppc64el) architectures.

If any email messages are falsely detected by the Dr.Web Anti-Spam component, we recommend that you forward them to special addresses for analysis and improvement of spam filter quality. To do that, save each message as a separate `.eml` file. Attach the saved files to an email message and forward it to the corresponding service address:

- nonspam@drweb.com—if it contains email files *erroneously classified as spam*;
- spam@drweb.com—if it contains spam email files *failed to be classified as spam*.

9.5.2. Command-Line Arguments

To start the Dr.Web Anti-Spam component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-ase [<parameters>]
```

Dr.Web Anti-Spam can process the following parameters:

Parameter	**Description**
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-ase --help
```

This command outputs short help information about the Dr.Web Anti-Spam component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon and controlled by the Dr.Web MailD component while scanning email messages for spam. At that, if a `FixedSocket` parameter value is set in the Dr.Web Anti-Spam component settings, then one Dr.Web Anti-Spam instance will be automatically started by the Dr.Web ConfigD component and will always be available to clients through the specified UNIX socket. To manage component parameters, as well as to scan mail objects on demand, use the Dr.Web Ctl tool designed for managing Dr.Web Security Space from the command line (the tool is run with the `drweb-ctl` command).

To scan an arbitrary email message with the Dr.Web Anti-Spam component for spam (by calling the Dr.Web MailD component), you can use the `checkmail` command of the Dr.Web Ctl tool. To do that, save the email message to be scanned to a disk (for example, in the `.eml` format) and run the command:

```
$ drweb-ctl checkmail <path to .eml file>
```

To get documentation for this component from the command line, run the `man 1 drweb-ase` command.

9.5.3. Configuration Parameters

The component uses configuration parameters specified in the `[Antispam]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, a value of the `DefaultLogLevel` parameter from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-ase`
`RunAsUser` *{UID | user name}*	The parameter defines a user on behalf of whom the component is run. Either the numerical UID of the user or the user name (login)

Parameter	Description
	can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the "`name:`" prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`FixedSocket` *{path to file}*	Path to a socket file of a fixed component instance. If this parameter is specified, the Dr.Web ConfigD configuration management daemon ensures that there is always a running component instance available to clients via this socket. Default value: *(not specified)*
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. If a `FixedSocket` value is set, this setting is ignored (the component does not finish its operation after the time interval expires). Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`FullCheck` *{logical}*	Perform or do not perform a full scan of the message for signs of spam. If `No`, the scanning will be stopped as soon as the spam scores exceeds the value specified in the `FastCheckStopThreshold` parameter. Default value: `Yes`
`FastCheckStopThreshold` *{integer}*	Spam score limit that, when reached, stops the message scan if the value of the `FullCheck` parameter is set to `No`. Default value: `300`
`AllowCyrillicText` *{logical}*	Increase or do not increase a spam score if the message has text in Cyrillic. If the parameter is set to `No`, then such message will have an increased spam score. Default value: `Yes`
`AllowCjkText` *{logical}*	Increase or do not increase a spam score if the message has text in Chinese, Korean or Japanese languages. If the parameter is set to `No`, then such message will have an increased spam score. Default value: `Yes`

Parameter	Description
`CheckCommercialEmails` *{logical}*	Exclude commercial messages (promotional emails, notifications of promotions and sales, and so on) from scanning. If `No`, such messages are classified as spam. Default value: `No`
`CheckSuspiciousEmails` *{logical}*	Exclude suspicious messages (for instance, those offering cash rewards) from scanning. If `No`, such messages are classified as spam. Default value: `No`
`CheckCommunityEmails` *{logical}*	Exclude social media messages from scanning. If `No`, such messages are classified as spam. Default value: `No`
`CheckTransactionalEmails` *{logical}*	Exclude transaction notifications (registration, purchase of services, goods and so on) from scanning. If `No`, such messages are classified as spam. Default value: `No`
`DetectSpamType` *{logical}*	Disable checking for a spam type (fraud or scamming). If `No`, such messages are classified as spam. Default value: `No`

9.6. Dr.Web Mail Quarantine

The Dr.Web Mail Quarantine message queue manager stores email messages and their metadata on the hard disk during the message scan.

Dr.Web Mail Quarantine is used by the Dr.Web MailD component while operating in SMTP or BCC mode. Dr.Web Mail Quarantine preserves message queues and ensures uninterrupted scanning and resending of messages in case of Dr.Web MailD errors or an MTA connection loss.

9.6.1. Operating Principles

The Dr.Web Mail Quarantine component serves two main purposes:

- Storing message queues and metadata on the hard disk during Dr.Web MailD message scans. Email messages are stored as files; their metadata is stored in an SQLite relational database. Storing of messages is required for SMTP and BCC modes.
- Redirection of a message to the Dr.Web MailD component after a certain time-out if an error occurs during the message processing (for instance, if the message could not be resent). The component ensures that the processed message gets delivered to the MTA.

The Dr.Web Mail Quarantine component cannot be started by the user in standalone mode. This component is started by the Dr.Web ConfigD configuration management daemon upon request of other components.

9.6.2. Command-Line Arguments

To start the Dr.Web Mail Quarantine component from the command line, run the following command:

```
$ drweb-ctl mailquarantine [<parameters>]
```

Dr.Web Mail Quarantine accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ drweb-ctl mailquarantine --help
```

This command outputs short help information about the Dr.Web Mail Quarantine component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To get documentation for this component from the command line, run the `man 1 drweb-mail-quarantine` command.

9.6.3. Configuration Parameters

The component uses configuration parameters specified in the `[MailQuarantine]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-mail-quarantine`
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the "`name:`" prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`SpoolDir` *{path to directory}*	Local file system directory used to store email messages and metadata. Default value: `/var/opt/drweb.com/lib/mail-quarantine`

9.7. SpIDer Gate

⚠ This component is included only in the distributions for GNU/Linux OSes.

The component for monitoring network traffic and URLs SpIDer Gate is designed to scan data downloaded from the network to a local computer or passed to the network from a local host for threats and to prevent connections to network hosts covered by the unwanted categories of web resources and by the black lists defined by the administrator.

Types of protocols for scanning can be indicated in the component settings. The component contains an analyzer of a protocol type used to send data via a monitored connection. If it is determined that the protocol is a mail one, data scan and threat search are performed by the Dr.Web MailD email message component.

To check whether a URL belongs to any of the categories (to scan connections that utilize the HTTP/HTTPS protocol), the component not only uses the database of web resource categories, which is updated regularly from the Doctor Web update servers, but also refers to the Dr.Web Cloud service. Doctor Web keeps track of the following web resources categories:

- *InfectionSource*—websites containing malware (“infection sources”).
- *NotRecommended*—fraudulent websites (that use “social engineering”) visiting which is not recommended.
- *AdultContent*—websites that contain pornographic or erotic materials, dating sites and so on.
- *Violence*—websites that encourage violence or contain materials about various fatal accidents and so on.
- *Weapons*—websites dedicated to weapons and explosives or providing information on their manufacturing and so on.
- *Gambling*—websites that provide access to online games of chance, casinos, auctions, including sites for placing bets and so on.
- *Drugs*—websites that promote use, production or distribution of drugs and so on.
- *ObsceneLanguage*—websites that contain obscene language (in section titles, articles and so on).
- *Chats*—websites that offer real-time exchange of text messages.
- *Terrorism*—websites that contain aggressive and propaganda materials or description of terrorist attacks, and so on.
- *FreeEmail*—websites that offer the possibility of free registration of an email box.
- *SocialNetworks*—social networking services: general, professional, corporate, interest-based; thematic dating websites.
- *DueToCopyrightNotice*—websites links to which are provided by the copyright holders of some copyrighted work (movies, music, and so on).
- *OnlineGames*—websites that provide access to games using a permanent internet connection.

- *Anonymizers*—websites allowing the user to hide personal information and providing access to blocked websites.
- *CryptocurrencyMiningPool*—websites that provide access to services for cryptocurrency mining.
- *Jobs*—job search websites.

Your system administrator can specify unwanted categories of hosts. Additionally, the user can configure their own black lists of hosts access to which will be blocked, and white lists of hosts access to which will be allowed even if they belong to the unwanted categories. If there is no information about URLs in the local black lists and the database of web resources categories, the component can send requests to the Dr.Web Cloud service to check whether these URLs are malicious. This information is received from other Dr.Web products on a real-time basis.

One and the same website can belong simultaneously to several categories. Access to such website is blocked if it belongs to any of the unwanted categories.

Even if a website is included in the white list, the data downloaded from the website or sent to it is scanned for threats.

If file scanning via HTTP is highly intensive, issues with scanning may arise when available file descriptors are depleted by the Dr.Web Network Checker component. In this case, it is necessary to increase the limit by the number of file descriptors available to Dr.Web Security Space.

9.7.1. Operating Principles

The SpIDer Gate component monitors network connections initiated by user applications. The component checks whether a server to which a client application is trying to connect belongs to any of the web resource categories specified in the settings as unwanted. Moreover, the component can use the Dr.Web Cloud service to scan URLs. If the URL belongs to any of the unwanted categories (or is flagged by the Dr.Web Cloud service) or to a black list defined by your system administrator, the connection is terminated and an HTML page with a message of that access is denied is displayed (in case of an HTTP/HTTPS connection). The page is generated by SpIDer Gate on the basis of a template supplied with the component. This page contains a notification of that access to the requested resource is impossible and describes a reason for blocking. A similar page is displayed and returned to the client if SpIDer Gate detects a threat that must be blocked in the data being transmitted. If the connection uses a protocol different from HTTP(S), the component only checks for permission to establish a connection with this server. If a mail protocol (SMTP, POP3 or IMAP) is used, the Dr.Web MailD component for scanning of email messages is used to analyze data and search for threats. This component parses email messages on its own and extracts attached files and URLs. At that, the component uses the same blocking parameters as the SpIDer Gate component.

The Dr.Web Firewall for Linux service component redirects connections to remote servers established by client applications transparently to them and exercises dynamic control of the rules of NetFilter, a system component of Linux.

The same Dr.Web Updater component regularly and automatically updates databases of web resource categories from Doctor Web servers and virus databases for Dr.Web Scanning Engine. The Dr.Web Cloud service is maintained by the Dr.Web CloudD component (using the cloud service is configured in the general settings of Dr.Web Security Space and can be disabled, if necessary). To scan data being transmitted, SpIDer Gate uses a network scanning agent, Dr.Web Network Checker, which initiates data scanning via Dr.Web Scanning Engine.

9.7.2. Command-Line Arguments

To start the SpIDer Gate component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-gated [<parameters>]
```

SpIDer Gate accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-gated --help
```

This command outputs short help information about the SpIDer Gate component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To get documentation for this component from the command line, run the `man 1 drweb-gated` command.

9.7.3. Configuration Parameters

The component uses configuration parameters specified in the `[GateD]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-gated`
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the “`name:`” prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will function indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`TemplatesDir` *{path to directory}*	Path to a directory that contains the templates for the HTML notifications sent upon blocking a web resource. Default value: `/var/opt/drweb.com/templates/gated`

Parameter	Description
`CaPath` *{path}*	Path to the directory or file with a list of trusted root certificates. Default value: *Path to the list of trusted certificates*. The path depends on your GNU/Linux distribution. • For Astra Linux, Debian, Linux Mint, SUSE Linux and Ubuntu this is usually the path `/etc/ssl/certs/`. • For CentOS and Fedora—`/etc/pki/tls/certs/ca-bundle.crt`. • The path can be defined for other distributions by running the `openssl version -d` command. • If the command is unavailable or your OS distribution cannot be identified, the `/etc/ssl/certs/` value is used.

⚠ Changes made to the settings of the connection scanning do not influence the scanning of connections that have already been established by the applications before making changes.

Other parameters of traffic monitoring, as well as its rules, are defined in the settings of the Dr.Web Firewall for Linux service component.

9.8. Dr.Web Firewall for Linux

⚠ This component is included only in the distributions designed for the OSes of the GNU/Linux family.

To enable the correct operation of the component, the OS kernel must be built with the following options:

- *CONFIG_NETLINK_DIAG*, *CONFIG_INET_TCP_DIAG*;
- *CONFIG_NF_CONNTRACK_IPV4*, *CONFIG_NF_CONNTRACK_IPV6*, *CONFIG_NF_CONNTRACK_EVENTS*;
- *CONFIG_NETFILTER_NETLINK_QUEUE*, *CONFIG_NETFILTER_NETLINK_QUEUE_CT*, *CONFIG_NETFILTER_XT_MARK*.

The set of required options from the specified list can depend on the GNU/Linux OS in use.

The Dr.Web Firewall for Linux is an auxiliary component functioning as a connection manager for SpIDer Gate. Dr.Web Firewall for Linux passes established connections through SpIDer Gate for scanning the transmitted traffic.

9.8.1. Operating Principles

In this section

- General Information
- Mechanism of Connection Interception
- Order of Connection Interception

General Information

The Dr.Web Firewall for Linux component ensures the correct operation of SpIDer Gate by analyzing routing rules adjusted for NetFilter (a system component of Linux) and modifies it so that the connections being established are redirected to SpIDer Gate, which functions as an intermediate (proxy) between a client application and a remote server.

Dr.Web Firewall for Linux can separately manage the rules for redirecting outgoing, incoming and transit connections. To configure the rules for passing or redirecting connections in detail, the component can use the rules incorporated in settings and a Lua script.

Mechanism of Connection Interception

To intercept connections, Dr.Web Firewall for Linux uses routing tables specified in the database of routing policies (see `man ip`: `ip route`, `ip rule`) and the *nf_conntrack* interface of the NetFilter system component. The intercepted connections and packets being transmitted are marked with bit marks to ensure proper routing thereby allowing Dr.Web Firewall for Linux to properly redirect connections and process packets being transmitted on various stages of passing NetFilter chains (for details, see `man iptables`).

Actions in iptables rules

Dr.Web Firewall for Linux uses the following actions in the `iptables` rules:

- *MARK*—assign a specified numeric mark to a packet.
- *CONNMARK*—assign a specified numeric mark to a connection.
- *TPROXY*—set an initial destination address of the connection and redirect packets from the *PREROUTING* NetFilter chain to a specified network socket (<*IP address*>:<*port*>) without changing contents of a packet.
- *NFQUEUE*—send a packet from the kernel network stack to a process that operates outside the kernel space for scanning. Dr.Web Firewall for Linux connects to the *NFQUEUE* queue with a specified number via a custom *Netlink* socket and receives packets for which a verdict on further processing must be reached (Dr.Web Firewall for Linux must inform NetFilter of one the following verdicts: *DROP*, *ACCEPT* or *REPEAT*).

Marks of packets and connections

To mark packets, Dr.Web Firewall for Linux uses the following three bits (out of available 32 bits) in packet and connection marks:

- `LDM` bit (*Local Delivery Mark*)—indicator of a local connection. Packets with this bit in the mark are sent to the local host using set routing rules.
- `CPM` bit (*Client Packets Mark*)—indicator of a connection between a client (a connection initiator) and a proxy, that is, Dr.Web Firewall for Linux.
- `SPM` bit (*Server Packets Mark*)—indicator of a connection between a proxy, that is, Dr.Web Firewall for Linux, and a server (a connection recipient).

`LDM`, `CPM` and `SPM` bits can be any *different* bits that are not used for marking packets by other applications that perform routing of connections. By default, Dr.Web Firewall for Linux chooses appropriate (not used by other applications) bits automatically.

Routes and routing policies (ip rule, ip route)

To ensure correct operation of Dr.Web Firewall for Linux (in any connection scanning mode), the `ip rule` routing policy that uses routing table #100 must be configured on your system:

```
from all fwmark <LDM>/<LDM> lookup 100
```

The following route must be added to this table:

```
local default dev lo scope host
```

This routing policy guarantees that packets with the `LDM` bit in their marks are always sent to the local host.

Hereinafter the expression <*XXX*> for the `XXX` bit is a *hexadecimal* value that equals 2^N, where *N* is a sequential number of the `XXX` bit in the packet mark. For example, if the lowest (zero) bit was selected as an `LDM` bit, then <*LDM*> = 2^0 = *0x1*.

NetFilter (iptables) rules

To ensure correct operation of Dr.Web Firewall for Linux (in any connection scanning mode), the following six rules (displayed in the `iptables-save` output command format) must be present in the `nat` and `mangle` tables of the corresponding chains of the NetFilter component:

```
*nat
-A POSTROUTING -o lo -m comment --comment drweb-firewall -m mark --mark
<LDM>/<LDM> -j ACCEPT
*mangle
-A PREROUTING -m comment --comment drweb-firewall -m mark --mark
0x0/<CPM+SPM> -m connmark --mark <SPM>/<CPM+SPM> -j MARK --set-xmark
<LDM>/<LDM>
-A PREROUTING -p tcp -m comment --comment drweb-firewall -m mark ! --mark
<CPM+SPM>/<CPM+SPM> -m connmark --mark <CPM>/<CPM+SPM> -j TPROXY --on-port
<port> --on-ip <IP address> --tproxy-mark <LDM>/<LDM>
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark
<CPM>/<CPM+SPM> -j CONNMARK --set-xmark <CPM>/0xffffffff
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark <SPM>/<CPM+SPM>
-j CONNMARK --set-xmark <SPM>/0xffffffff
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark 0x0/<CPM+SPM> -
m connmark ! --mark 0x0/<CPM+SPM> -j MARK --set-xmark <LDM>/<LDM>
```

These rules are numbered 0–5 in the description below (in the order they are listed here). The expression <*X*+*Y*> means a number equal to bitwise “OR” (a sum) of the corresponding numbers *X* and *Y*.

Parameters <*IP address*> and <*port*> in rule #2 specify a network socket on which Dr.Web Firewall for Linux manages intercepted connections.

Furthermore, the following additional rules (per rule for each of the modes) must be present in `mangle` tables of the corresponding chains (*OUTPUT*, *INPUT*, *FORWARD*) when enabling the interception mode for outgoing, incoming and transit connections in the Dr.Web Firewall for Linux settings:

- To intercept outgoing connections (*OUTPUT*):

```
-A OUTPUT -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <ONum> --
queue-bypass
```

- To intercept incoming connections (*INPUT*):

```
-A INPUT -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <INum> --
queue-bypass
```

- To intercept transit connections (*FORWARD*):

```
-A FORWARD -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <FNum> --
queue-bypass
```

> These rules are numbered 6, 7 and 8 in the description below (in the order they are listed here).

Here, <*ONum*>, <*INum*> and <*FNum*> are numbers of queues in *NFQUEUE*, in which Dr.Web Firewall for Linux is waiting for packets that indicate establishing connections with the corresponding directions (these are packets with a set `SYN` flag and an unset `ACK` flag).

Order of Connection Interception

In accordance with any of rules 6, 7 and 8, packets indicating a new network connection of the corresponding direction, if not marked by both `CPM` and `SPM` bits, are put in the corresponding *NFQUEUE* queues by NetFilter, where they will be read by Dr.Web Firewall for Linux via the *nf_conntrack* interface. Rules 3 and 4 mark the connection itself as intercepted, that is, a bit indicating the connection direction is set in the connection mark. This bit number matches the bit number in the packet mark. As the result, in accordance with rules 1, 2 and 5, Dr.Web Firewall for Linux will receive packets sent via this connection. Rule 0 is added on top of the *POSTROUTING* chain in the `nat` table, so that if NAT is configured, not to translate addresses for marked packets, because this will interfere with Dr.Web Firewall for Linux logic of connection interception and processing.

When a packet appears in one of the *NFQUEUE* queues, Dr.Web Firewall for Linux performs basic verification of the packet in the event invalid rules are set in NetFilter. Subsequently, Dr.Web Firewall for Linux attempts to connect on behalf of itself to the server from the socket marked with `PSC`. At that, rule 4 is triggered. Rule 5 for local delivery is not triggered, because the packet is marked with `SPM` and this rule is only applicable to packets marked with <*CPM+SPM*>.

- *If the connection to the server fails*, Dr.Web Firewall for Linux generates a client packet with an `RST` bit having replaced the pair <*IP address*>:<*port*> with the address of the network socket of the requested server. At that, the *DROP* verdict is sent to *NFQUEUE*. The socket used for sending the packet with the `RST` bit is marked as <*CPM+SPM*>, so none of the above rules is triggered, and this packet will be delivered to the client in accordance with standard routing rules.
- *If an attempt to connect to a remote server is successful*, Dr.Web Firewall for Linux copies the intercepted SYN packet and resends it from the socket marked as <*LDM+CPM*> to redirect

the sent packet to the local network socket. Owing to the set `LDM` bit and in accordance with the specified routing rules, when selecting an output interface, the packet will be sent to the *loopback* interface and then to the NetFilter *PREROUTING* chain, where rule 2 will be triggered. Thus, the packet will be redirected to the network socket of Dr.Web Firewall for Linux unchanged. This method allows Dr.Web Firewall for Linux to preserve all four elements of the connection address (an IP address and a port of a packet sender, an IP address and a port of a packet recipient).

The `IP_TRANSPARENT` option and the *<LDM+CPM>* mark are set for the network socket on which Dr.Web Firewall for Linux receives intercepted connections in accordance with rule 2; therefore, packets sent by Dr.Web Firewall for Linux from this socket are not included in the *NFQUEUE* queues. When a client connects, search is performed for a paired socket using the preserved four-element address (the IP address and the port of the packet sender, the IP address and the port of the packet recipient). When the connection with the client and the server is established, it is scanned using a Lua procedure, as well as scanning rules specified in the Dr.Web Firewall for Linux settings. If the scans are successful and the connection should not be closed, the associated socket pair that connects the client and server sides of the established connection is passed to the SpIDer Gate component for analyzing the data transmitted over the connection. The further interaction of the client and the server is mediated by SpIDer Gate. In addition to the socket pair associated with the client and server sides of the connection, SpIDer Gate receives the parameters and rules for scanning the established connection from Dr.Web Firewall for Linux.

Simplified diagram of Dr.Web Firewall for Linux operation is provided below.

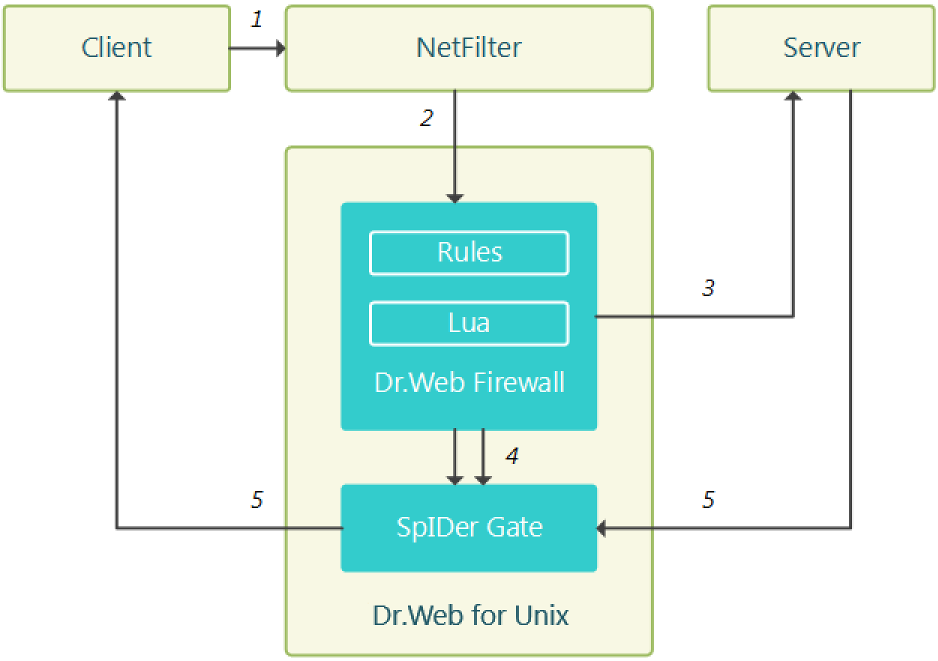


Figure 52. Diagram of Dr.Web Firewall for Linux operation

The following connection processing steps are numbered:

1. Client attempts to connect to the server.

2. NetFilter redirects the connection being established to Dr.Web Firewall for Linux in accordance with the routing rules.
3. Dr.Web Firewall for Linux attempts to connect to the server on behalf of the client; the connection is scanned.
4. Socket pair associated with the client and server sides of the connection is passed to SpIDer Gate for processing the connection along with the parameters and rules for scanning it.
5. Server and the client exchange data via SpIDer Gate that serves as a mediator.

⚠ For correct operation of Dr.Web Firewall for Linux, these rules must be present in routing tables with correct numbers of bit marks, *NFQUEUE* queues and network socket addresses for connection interception. By default, the component performs the necessary configuration of rules automatically. If automatic configuration of connections is disabled in the component settings, it is necessary to provide the required rules manually when starting the component.

9.8.2. Command-Line Arguments

To start the Dr.Web Firewall for Linux component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-firewall [<parameters>]
```

Dr.Web Firewall for Linux accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-firewall --help
```

This command outputs short help information about the Dr.Web Firewall for Linux component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

> To start the Dr.Web Ctl tool, run the `drweb-ctl` command.
>
> To get documentation for this component from the command line, run the `man 1 drweb-firewall` command.

9.8.3. Configuration Parameters

In this section

- Component Parameters
- Rules for Traffic Monitoring and Blocking of Access

The component uses configuration parameters specified in the `[LinuxFirewall]` section of the unified configuration file of Dr.Web Security Space.

Component Parameters

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-firewall`
`XtablesLockPath` *{path to file}*	Path to the `iptables` (NetFilter) table blocking file. If the parameter value is not specified, the `/run/xtables.lock`

Parameter	Description
	and `/var/run/xtables.lock` paths are checked. If the file is not found in the specified path or default paths, an error occurs upon starting the component. Default value: *(not specified)*
`InspectFtp` *{On | Off}*	Scan the data transferred over the FTP protocol. The data will be scanned on the basis of the specified rules (see below). Default value: `On`
`InspectHttp` *{On | Off}*	Scan the data transferred over the HTTP protocol. The data will be scanned on the basis of the specified rules (see below). Default value: `On`
`InspectSmtp` *{On | Off}*	Scan the data transferred over the SMTP protocol (the Dr.Web MailD component is used). The data will be scanned on the basis of the specified rules (see below). Default value: `On`
`InspectPop3` *{On | Off}*	Scan the data transferred over the POP3 protocol (the Dr.Web MailD component is used). The data will be scanned on the basis of the specified rules (see below). Default value: `On`
`InspectImap` *{On | Off}*	Scan the data transferred over the IMAP protocol (the Dr.Web MailD component is used). The data will be scanned on the basis of the specified rules (see below). Default value: `On`
`AutoconfigureIptables` *{Yes | No}*	Enable or disable the mode of configuring the rules for the NetFilter system component via the `iptables` interface. Allowed values: • `Yes`—automatically configure NetFilter rules upon starting the component and remove them upon finishing its operation (*recommended*). • `No`—do not configure the rules automatically. The required rules must be added manually by the

Parameter	Description
	administrator before starting the component and removed after it finishes its operation. ⚠ If the automatic configuration of the `iptables` rules is not allowed, the required `iptables` rules must be ensured before the component operation starts. Default value: `Yes`
`AutoconfigureRouting` *{Yes | No}*	The configuration mode for `ip route` and `ip rule` routing rules and policies. Allowed values: • `Yes`—automatically configure rules and `ip route` and `ip rule` routing policies upon starting the component and remove them upon finishing its operation (*recommended*). • `No`—do not configure the rules automatically. The required rules must be added manually by the administrator before starting the component and removed after it finishes its operation. ⚠ If the automatic configuration of the routing rules and policies is not allowed, the required rules for `ip route` and `ip rule` must be available before the component operation starts. Default value: `Yes`
`LocalDeliveryMark` *{integer | Auto}*	The *<LDM>* mark for packets that are redirected to the Dr.Web Firewall for Linux network socket (specified in the `TproxyListenAddress` parameter, see below) to intercept the connection. Allowed values: • *<integer>*—*<LDM>* mark for packets. Equals 2^N, where *N* is an `LDM` bit number in the packet, $0 \le N \le 31$. • `Auto`—allow Dr.Web Firewall for Linux to select the appropriate bit in the packet mark automatically (*recommended*).

Parameter	Description
	⚠ When assigning <*LDM*> number manually, make sure that the corresponding bit in the packet mark is not used by any other applications that route connections and packets (including via NetFilter). If an invalid value is specified, the component will fail to start. The specified <*LDM*> number must be used in the routing rules to be added manually, if `AutoconfigureIptables = No` and/or `AutoconfigureRouting = No`. Default value: `Auto`
`ClientPacketsMark` *{integer | Auto}*	The <*CPM*> mark for packets transferred between the client that initiates the connection and Dr.Web Firewall for Linux. Allowed values: • <*integer*>—<*CPM*> mark for packets. Equals 2^N, where *N* is an `CPM` bit number in the packet, 0 ≤ *N* ≤ 31. • `Auto`—allow Dr.Web Firewall for Linux to select the appropriate bit in the packet mark automatically (*recommended*). ⚠ When assigning the <*CPM*> number manually, make sure that the corresponding bit in the packet mark is not used by any other applications that route connections and packets (including via NetFilter). If an invalid value is specified, the component will fail to start. The specified <*CPM*> number must be used in the routing rules to be added manually, if `AutoconfigureIptables = No`. Default value: `Auto`
`ServerPacketsMark` *{integer | Auto}*	<*SPM*> mark for packets transferred between Dr.Web Firewall for Linux and the server that receives the

Parameter	Description
	connection. Allowed values: • *<integer>*—*<SPM>* mark for packets. Equals 2^N, where *N* is an `SPM` bit number in the packet, $0 \le N \le 31$. • `Auto`—allow Dr.Web Firewall for Linux to select the appropriate bit in the packet mark automatically (*recommended*). ⚠ When assigning the *<SPM>* number manually, make sure that the corresponding bit in the packet mark is not used by any other applications that route connections and packets (including via NetFilter). If an invalid value is specified, the component will fail to start. The specified *<SPM>* number must be used in the routing rules to be added manually, if `AutoconfigureIptables = No` and/or `AutoconfigureRouting = No`. Default value: `Auto`
`TproxyListenAddress` *{network socket}*	Network socket (*<IP address>*:*<port>*) on which Dr.Web Firewall for Linux receives intercepted connections. If you specify a zero port number, the socket is selected automatically by the system. ⚠ It is necessary to make sure that the corresponding socket is not used by any other applications. If an invalid value is specified, the component will fail to start. The specified IP address and port must be used in the routing rules to be added manually, if `AutoconfigureIptables = No`. Default value: `127.0.0.1:0`
`OutputDivertEnable` *{Yes | No}*	Enable or disable the interception mode for incoming connections (i.e. connections initiated by applications on

Parameter	Description
	the local host). Allowed values: • `Yes`—intercept and process outgoing connections; • `No`—do not intercept or process outgoing connections. ⚠ This setting adds or removes routing rule #5 to be added or removed manually if `AutoconfigureIptables = No`. Default value: `No`
`OutputDivertNfqueueNumber` *{integer | Auto}*	Number of the *NFQUEUE* queue from which Dr.Web Firewall for Linux will retrieve SYN packets that initiate outgoing connections. Allowed values: • <*integer*>—<*ONum*> queue number to monitor the SYN packets of intercepted outgoing connections in *NFQUEUE*; • `Auto`—allow Dr.Web Firewall for Linux to select an appropriate queue number automatically (*recommended*). ⚠ When assigning the <*ONum*> number manually, make sure that the corresponding queue is not used by any other applications that control connections and packets (including via the NetFilter rules). If an invalid value is specified, the component will fail to start. The specified <*ONum*> number must be used in routing rule #5 to be added manually if `AutoconfigureIptables = No`. Default value: `Auto`
`OutputDivertConnectTransparently` *{Yes | No}*	Enable or disable the emulation mode for connecting to the recipient (server) using the IP address of the sender of an intercepted packet (client) for outgoing connections.

Parameter	Description
	Allowed values: • `Yes`—connect to the server using the address of the client that requested the connection instead of your own when intercepting the connection; • `No`—connect to the server from the Dr.Web Firewall for Linux address. As the client and Dr.Web Firewall for Linux addresses are usually the same in the outgoing connection interception mode, the default value is `No`. Default value: `No`
`InputDivertEnable` *{Yes | No}*	Enable or disable interception of incoming connections (i.e. connections initiated by applications on the remote host to applications operating on the local host). Allowed values: • `Yes`— intercept and process incoming connections; • `No`—do not intercept or process outgoing connections. ⚠ This setting adds or removes routing rule #6 to be added or removed manually if `AutoconfigureIptables = No`. If an invalid value is specified, the component will fail to start. Default value: `No`
`InputDivertNfqueueNumber` *{integer | Auto}*	Number of the *NFQUEUE* queue from which Dr.Web Firewall for Linux will retrieve SYN packets that initiate incoming connections. Allowed values: • *<integer>*—*<INum>* queue number to monitor the SYN packets of intercepted incoming connections in *NFQUEUE*; • `Auto`—allow Dr.Web Firewall for Linux to select an appropriate queue number automatically (*recommended*).

Parameter	Description
	⚠ When assigning the *<INum>* number manually, make sure that the corresponding queue is not used by any other applications that control connections and packets (including via the NetFilter rules). If an invalid value is specified, the component will fail to start. The specified *<INum>* number must be used in routing rule #6 to be added manually if `AutoconfigureIptables = No`. Default value: `Auto`
`InputDivertConnectTransparently` *{Yes | No}*	Enable or disable the emulation mode for connecting to the recipient (server) using the IP address of the sender of an intercepted packet (client) for incoming connections. Allowed values: • `Yes`—connect to the server using the address of the client that requested the connection instead of your own when intercepting the connection; • `No`—connect to the server from the Dr.Web Firewall for Linux address. In the incoming connection interception mode, all traffic goes through Dr.Web Firewall for Linux, and it is possible to connect safely to the server using the fake client address; therefore, the default value is `Yes`. Default value: `Yes`
`ForwardDivertEnable` *{Yes | No}*	Enable or disable interception of transit connections (i.e. connections initiated by applications on one remote host to applications on another remote host). Allowed values: • `Yes`—intercept and process transit connections; • `No`—do not intercept or process transit connections. ⚠ This setting adds or removes routing rule #7 to be added or removed manually if `AutoconfigureIptables = No`.

Parameter	Description
	Default value: `No`
`ForwardDivertNfqueueNumber` *{integer | Auto}*	Number of the *NFQUEUE* queue from which Dr.Web Firewall for Linux will retrieve SYN packets that initiate transit connections. Allowed values: • <*integer*>—<*FNum*> queue number to monitor the SYN packets of intercepted transit connections in *NFQUEUE*; • `Auto`—allow Dr.Web Firewall for Linux to select an appropriate queue number automatically (*recommended*). ⚠ When assigning the <*FNum*> number manually, make sure that the corresponding queue is not used by any other applications that control connections and packets (including via the NetFilter rules). If an invalid value is specified, the component will fail to start. The specified <*FNum*> number must be used in routing rule #7 to be added manually if `AutoconfigureIptables = No`. Default value: `Auto`
`ForwardDivertConnectTransparently` *{Yes | No}*	Enable or disable the emulation mode for connecting to the recipient (server) using the IP address of the sender of an intercepted packet (client) for transit connections. Allowed values: • `Yes`—connect to the server using the address of the client that requested the connection instead of your own when intercepting the connection; • `No`—connect to the server from the Dr.Web Firewall for Linux address. As there is no guarantee that in the transit connection interception mode all the traffic goes through the same host (router) on which Dr.Web Firewall for Linux is installed, to ensure the correct operation, the default value is `No`. If the network configuration guarantees that protected applications use the same router, the parameter can be set to `Yes`, and in this case, Dr.Web

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td></td><td>

Firewall for Linux will always emulate the connection from the client address when connecting to servers.

Default value: `No`

</td></tr>
<tr><td>

`ExcludedProc`

{path to file}

</td><td>

White list of processes (the network activity of which is not controlled).

Multiple values can be specified as a list. List values must be comma-separated and put in quotation marks. The parameter can be specified more than once in the section (in this case, all parameter values are combined into one list).

Example: Add the `wget` and `curl` processes to the list.

1. Adding values to the configuration file.
   - Two values per line:

```
[LinuxFirewall]
ExcludedProc = "/usr/bin/wget",
"/usr/bin/curl"
```

   - Two lines (one value per line):

```
[LinuxFirewall]
ExcludedProc = /usr/bin/wget
ExcludedProc = /usr/bin/curl
```

2. Adding values with the `drweb-ctl cfset` command:

```
# drweb-ctl cfset
LinuxFirewall.ExcludedProc -
a /usr/bin/wget
# drweb-ctl cfset
LinuxFirewall.ExcludedProc -
a /usr/bin/curl
```

⚠ Actual usage of the process list indicated in this parameter depends on the *method* of its usage in the scanning rules defined for Dr.Web Firewall for Linux.

The list of default rules (see below) guarantees that traffic of all processes from the list is allowed *without any scanning*.

Default value: *(not specified)*

</td></tr>
</table>

Parameter	Description
`UnwrapSsl` *{boolean}*	Scan or do not scan encrypted traffic passing via SSL. In the current implementation, the value if this variable does not influence scanning of secure traffic. To control scanning, it is necessary to create a rule comprising the action `SET Unwrap_SSL = true/false` (see below). If you change the value of this parameter using the `cfset` command of the `drweb-ctl` utility, the affected dependent rules will adapt automatically. Default value: `No`
`BlockInfectionSource` *{boolean}*	Block attempts to connect to websites containing malware (belonging to the *InfectionSource* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `Yes`
`BlockNotRecommended` *{boolean}*	Block attempts to connect to non-recommended websites (belonging to the *NotRecommended* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `Yes`
`BlockAdultContent` *{boolean}*	Block attempts to connect to websites containing adult content (belonging to the *AdultContent* category).

Parameter	Description
	To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockViolence` *{boolean}*	Block attempts to connect to websites containing graphic violence (belonging to the *Violence* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockWeapons` *{boolean}*	Block attempts to connect to websites dedicated to weapons (belonging to the *Weapons* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockGambling` *{boolean}*	Block attempts to connect to gambling websites (belonging to the *Gambling* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockDrugs` *{boolean}*	Block attempts to connect to websites dedicated to drugs (belonging to the *Drugs* category).

Parameter	Description
	To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockObsceneLanguage` *{boolean}*	Block attempts to connect to websites with obscene language (belonging to the *ObsceneLanguage* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockChats` *{boolean}*	Block attempts to connect to chat websites (belonging to the *Chats* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockTerrorism` *{boolean}*	Block attempts to connect to websites dedicated to terrorism (belonging to the *Terrorism* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockFreeEmail` *{boolean}*	Block attempts to connect to websites of free email services (belonging to the *FreeEmail* category).

Parameter	Description
	To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockSocialNetworks` *{boolean}*	Block attempts to connect to social networking websites (belonging to the *SocialNetworks* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockDueToCopyrightNotice` *{boolean}*	Block attempts to connect to websites that were added at the request of a copyright holder (belonging to the *DueToCopyrightNotice* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `Yes`
`BlockOnlineGames` *{boolean}*	Block attempts to connect to online gaming websites (belonging to the *OnlineGames* category). To enable blocking, the settings must contain the following rule (see below): `url_category in "LinuxFirewall.BlockCategory" : BLOCK as _match` Default value: `No`
`BlockAnonymizers` *{boolean}*	Block attempts to connect to anonymizer websites (belonging to the *Anonymizers* category).

Parameter	Description

	To enable blocking, the settings must contain the following rule (see below):
	`url_category in`
	`"LinuxFirewall.BlockCategory" : BLOCK`
	`as _match`
	Default value: `No`

`BlockCryptocurrencyMiningPools`
{boolean}
	Block attempts to connect to websites combining users to mine cryptocurrencies (belonging to the *CryptocurrencyMiningPool* category).
	To enable blocking, the settings must contain the following rule (see below):
	`url_category in`
	`"LinuxFirewall.BlockCategory" : BLOCK`
	`as _match`
	Default value: `No`

`BlockJobs`
{boolean}
	Block attempts to connect to job search websites (belonging to the *Jobs* category).
	To enable blocking, the settings must contain the following rule (see below):
	`url_category in`
	`"LinuxFirewall.BlockCategory" : BLOCK`
	`as _match`
	Default value: `No`

`Whitelist` *{domain list}*	White list of domains (domains allowed for connection, even if these domains belong to blocked categories of web resources. In addition, user access is allowed to all subdomains of the domains from this list). The values of the list must be comma-separated (each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all parameter values are combined into one list). Example: Add the `example.com` and `example.net` domains to the list. 1. Adding values to the configuration file.

Parameter	Description

	- Two values per line:
	`[LinuxFirewall]`
	`Whitelist = "example.com", "example.net"`
	- Two lines (one value per line):
	`[LinuxFirewall]`
	`Whitelist = example.com`
	`Whitelist = example.net`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset LinuxFirewall.Whitelist -a example.com`
	`# drweb-ctl cfset LinuxFirewall.Whitelist -a example.net`
	⚠ Actual usage of the domain list indicated in this parameter depends on the *method* of its usage in the scanning rules defined for Dr.Web Firewall for Linux.
	The list of default rules (see below) guarantees that access to domains (and their subdomains) from this list will be provided even if it contains domains from the list of blocked web resource categories, but only in case of a request to a host via the HTTP protocol. Moreover, the default set of rules guarantees that data downloaded from the white list of domains *will be scanned for threats* (because the data is returned in a response, and the `direction` variable has the `response` value).
	Default value: *(not specified)*

`Blacklist` *{domain list}*	Black list of domains (i.e. the domains forbidden for connection, even if these domains do not belong to blocked categories of web resources. In addition, user access will be forbidden to all subdomains of the domains from this list).

Parameter	Description

	The values of the list must be comma-separated (each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all parameter values are combined into one list).
	Example: Add the `example.com` and `example.net` domains to the list.
	1. Adding values to the configuration file.
	• Two values per line:
	`[LinuxFirewall]`
	`Blacklist = "example.com", "example.net"`
	• Two lines (one value per line):
	`[LinuxFirewall]`
	`Blacklist = example.com`
	`Blacklist = example.net`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset LinuxFirewall.Blacklist -a example.com`
	`# drweb-ctl cfset LinuxFirewall.Blacklist -a example.net`
	⚠ Actual usage of the domain list indicated in this parameter depends on the *method* of its usage in the scanning rules defined for Dr.Web Firewall for Linux.
	The list of default rules (see below) guarantees that access to the domains (and their subdomains) from this list via the HTTP protocol will always be forbidden. If a domain is added simultaneously to `Whitelist` and `Blacklist`, the default rules guarantee that user access to the domain via the HTTP protocol will be blocked.
	Default value: *(not specified)*

Parameter	Description
`ScanTimeout` *{time interval}*	Timeout for scanning one file at the request of SpIDer Gate. Allowed values: from 1 second (`1s`) to 1 hour (`1h`). Default value: `30s`
`HeuristicAnalysis` *{On | Off}*	Enable or disable the heuristic analysis for detection of unknown threats during file scanning initiated by SpIDer Gate. The heuristic analysis provides higher detection reliability, but, at the same time, increases scanning time. Action applied to threats detected by the heuristic analyzer is specified as the `BlockSuspicious` parameter value. Allowed values: • `On`—enable the heuristic analysis while scanning; • `Off`—disable the heuristic analysis. Default value: `On`
`PackerMaxLevel` *{integer}*	Maximum nesting level for packed objects. A packed object is executable code compressed with special software (UPX, PELock, PECompact, Petite, ASPack, Morphine and so on). Such objects may include other packed objects that may also include packed objects and so on. The value of this parameter specifies the nesting limit beyond which packed objects inside other packed objects are not scanned. The nesting level is not limited. If the value is set to 0, nested objects are not scanned. Default value: `8`
`ArchiveMaxLevel` *{integer}*	Maximum nesting level for archives (`.zip`, `.rar` and so on) in which other archives may be enclosed, whereas these archives may also include other archives and so on. The value of this parameter specifies the nesting limit beyond which archives enclosed in other archives are not scanned. The nesting level is not limited. If the value is set to 0, nested objects are not scanned. Default value: `8`
`MailMaxLevel` *{integer}*	Maximum nesting level for mailer files (`.pst`, `.tbb` and so on) in which other files may be enclosed, whereas these files may also include other files and so on. The

Parameter	Description
	value of this parameter specifies the nesting limit beyond which objects inside other objects are not scanned. The nesting level is not limited. If the value is set to 0, nested objects are not scanned. Default value: `8`
`ContainerMaxLevel` *{integer}*	Maximum nesting for other types objects inside which other objects are enclosed (HTML pages, `.jar` files and so on). The value of this parameter specifies the nesting limit beyond which objects inside other objects will not be scanned by the request of SpIDer Gate. The nesting level is not limited. If the value is set to `0`, nested objects are not scanned. Default value: `8`
`MaxCompressionRatio` *{integer}*	Maximum compression ratio of compressed/packed objects (ratio between the uncompressed size and the compressed size). If the ratio of an object exceeds the limit, this object will be skipped during file scanning initiated by SpIDer Gate. The compression ratio must be no less than `2`. Default value: `500`
`BlockKnownVirus` *{boolean}*	Block incoming and outgoing data if it contains a known threat. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `Yes`
`BlockSuspicious` *{boolean}*	Block incoming and outgoing data if it contains an unknown threat detected by the heuristic analyzer. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `Yes`

Parameter	Description
`BlockAdware` *{boolean}*	Block incoming and outgoing data if it contains adware. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `Yes`
`BlockDialers` *{boolean}*	Block incoming and outgoing data if it contains a dialer. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `Yes`
`BlockJokes` *{boolean}*	Block incoming and outgoing data if it contains a joke program. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `No`
`BlockRiskware` *{boolean}*	Block incoming and outgoing data if it contains riskware. To enable blocking, the settings must contain the following rule (see below): `threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match` Default value: `No`
`BlockHacktools` *{boolean}*	Block incoming and outgoing data if it contains a hack tool.

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td></td><td>

To enable blocking, the settings must contain the following rule (see below):

```
threat_category in
"LinuxFirewall.BlockThreat" : BLOCK as
_match
```

Default value: `No`

</td></tr>
<tr><td>

`BlockUnchecked`

{boolean}

</td><td>

Block incoming and outgoing data if it cannot be scanned.

The value of this parameter influences processing of the rules that are impossible to evaluate to true or false because of an error. If `No` is specified, the rule is skipped as the rule that has not been executed. If `Yes` is specified, the `BLOCK as BlackList` action is performed.

Default value: `No`

</td></tr>
<tr><td>

`InterceptHook`

{path to file | Lua function}

</td><td>

Script for processing connections in Lua or a path to the file storing this script.

If the specified file is unavailable, starting the component causes an error.

Default value:

```
local dwl = require 'drweb.lookup'

function intercept_hook(ctx)

  -- do not check if group ==
Root.TrustedGroup
  if ctx.divert == "output" and
ctx.group == "drweb"
  then
      return "pass"
  end

  -- do not check connections from
privileged ports
  -- except FTP active mode
  if ctx.src.port >= 0 and ctx.src.port
<= 1024
      and ctx.src.port ~= 20
  then
```

</td></tr>
</table>

Parameter	Description

	`    return "pass"`
	`  end`
	
	`  return "check"`
	`end`

> Changes made to the settings of the connection scanning do not influence the scanning of connections that have already been established by applications before making changes. If it is required to apply them to already running applications, it is necessary to force them to disconnect and then connect again, for example, by restarting these applications.

Rules for Traffic Monitoring and Blocking of Access

In addition to the parameters listed above, the section also contains 11 *sets of rules* `RuleSet*` (`RuleSet0`, ..., `RuleSet10`) which directly control traffic scanning and blocking of user access to web resources, as well as blocking downloading content from the internet. For some values in conditions (for example, IP address ranges, lists of website categories, black and white lists of web resources, etc.), there is a substitution of values loaded from text files and also extracted from external data sources via LDAP. When configuring connections, all rules are checked from first to last as a single list, until the rule that triggered the ultimate resolution is found. The gaps in the rule list, if any, are ignored.

Viewing and editing the rules

List gaps, i.e. `RuleSet`<*i*> sets that do not contain rules (wherein <*i*> is a `RuleSet` rule set number), are kept for editing the rules easily.

> You *cannot* add items different to the already present `RuleSet`<*i*> items, but you can add or remove any rule in any `RuleSet`<*i*> item.

View and edit rules in any of the following ways:

- by viewing and editing the configuration file (in any text editor) (note that this file stores only those parameters that values are different from the defaults);
- via the command-line interface—Dr.Web Ctl (`drweb-ctl cfshow` and `drweb-ctl cfset` commands).

> If you have edited the rules and made changes to the configuration file, in order to apply these changes, restart Dr.Web Security Space. To do that, use the `drweb-ctl reload` command.

Viewing the rules with the drweb-ctl cfshow command

To view the contents of the `LinuxFirewall.RuleSet1` rule set, use the command:

```
# drweb-ctl cfshow LinuxFirewall.RuleSet1
```

Editing the rules using the drweb-ctl cfset command (hereinafter <*rule*> is text of the rule):

- Replace all the rules in the `LinuxFirewall.RuleSet1` set with a new rule:

  ```
  # drweb-ctl cfset LinuxFirewall.RuleSet1 '<rule>'
  ```

- Add a new rule to the `LinuxFirewall.RuleSet1` rule set:

  ```
  # drweb-ctl cfset -a LinuxFirewall.RuleSet1 '<rule>'
  ```

- Remove a specific rule from the `LinuxFirewall.RuleSet1` rule set:

  ```
  # drweb-ctl cfset -e LinuxFirewall.RuleSet1 '<rule>'
  ```

- Reset the `LinuxFirewall.RuleSet1` rule set to the default state:

  ```
  # drweb-ctl cfset -r LinuxFirewall.RuleSet1
  ```

When you use the `drweb-ctl` tool to edit the rules, put the text of the <*rule*> rule to be added in single or double quotes and use a backward slash (`\`) to escape quotes within the text of the rule.

It is important to remember the following aspects of storing rules in the `RuleSet`<*i*> configuration variables:

- The conditional part and colon can be omitted when adding unconditional rules. However, such rules are always stored in the list of rules as the '` : `<*action*>' string.
- When adding rules that contain several actions (such rules as '<*condition*>` : `<*action 1*>`, `<*action 2*>'), such rules will be transformed into a chain of elementary rules '<*condition*>` : `<*action 1*>' and '<*condition*>` : `<*action 2*>'.
- Rules do not allow for disjunction (logical “OR”) of conditions in the conditional part, so, in order to implement the logical “OR”, construct a chain of rules with each rule having a disjunct-condition in its condition.

To add an unconditional skipping rule (the `PASS` action) to the `LinuxFirewall.RuleSet1` rule set, run the following command:

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'PASS'
```

To remove this rule from the specified rule set, run the following command:

```
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 ' : PASS'
```

To add a rule to change a path to standard templates for connections from forbidden addresses and block these connections to the `LinuxFirewall.RuleSet1` rule set, run the command:

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : set http_template_dir = "mytemplates", BLOCK'
```

This command adds *two rules* to the specified rule set, so, in order to remove these rules, run these two commands:

```
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : set http_template_dir = "mytemplates"'
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : BLOCK'
```

To add such rule as “Block upon detecting a malicious object of the *KnownVirus* type or a URL from the *Terrorism* category” to the `LinuxFirewall.RuleSet1` rule set, add the following two rules:

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'threat_category in (KnownVirus)
: BLOCK as _match'
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'url_category in (Terrorism) :
BLOCK as _match'
```

To remove them, run two commands, as shown in the example above.

Default set of rules

By default, the following set of rules for blocking is specified:

```
RuleSet0 =
RuleSet1 = divert output : set HttpTemplatesDir = "output"
RuleSet1 = divert output : set MailTemplatesDir = "firewall"
RuleSet1 = divert input : set HttpTemplatesDir = "input"
RuleSet1 = divert input : set MailTemplatesDir = "server"
RuleSet1 = proc in "LinuxFirewall.ExcludedProc" : PASS
RuleSet1 =  : set Unwrap_SSL = false
RuleSet2 =
RuleSet3 =
RuleSet4 =
RuleSet5 = protocol in (Http), direction request, url_host in
"LinuxFirewall.Blacklist" : BLOCK as BlackList
RuleSet5 = protocol in (Http), direction request, url_host in
"LinuxFirewall.Whitelist" : PASS
RuleSet6 =
RuleSet7 = protocol in (Http), direction request, url_category in
"LinuxFirewall.BlockCategory" : BLOCK as _match
RuleSet8 =
RuleSet9 = protocol in (Http), divert input, direction request,
threat_category in "LinuxFirewall.BlockThreat" : BLOCK as _match
RuleSet9 = protocol in (Http), direction response, threat_category in
"LinuxFirewall.BlockThreat" : BLOCK as _match
```

```
RuleSet9 = protocol in (Smtp), threat_category in
"LinuxFirewall.BlockThreat" : REJECT
RuleSet9 = protocol in (Smtp), url_category in "LinuxFirewall.BlockCategory"
: REJECT
RuleSet9 = protocol in (Smtp), total_spam_score gt 0.80 : REJECT
RuleSet9 = protocol in (Pop3, Imap), threat_category in
"LinuxFirewall.BlockThreat" : REPACK as _match
RuleSet9 = protocol in (Pop3, Imap), url_category in
"LinuxFirewall.BlockCategory" : REPACK as _match
RuleSet9 = protocol in (Pop3, Imap), total_spam_score gt 0.80 : REPACK as
_match
RuleSet10 =
```

The first rule indicates that if the connection is established by the process specified in the `ExcludedProc` parameter (see above), the connection is skipped without checking any other conditions. The next rule (executed without any condition) disables scanning of secure connections. This and all the rules below are analyzed only if the connection is not associated with the excluded process. Moreover, as all subsequent rules depend on the protocol type, if scanning of secure connections is disabled and the connection is secure, the rules are not executed because it is impossible to define whether the conditions are true.

The following five rules regulate processing of outgoing HTTP connections:

1. If the host to which a connection is established is included in a black list, the connection is blocked without performing other checks.
2. If the host is included in a white list, the connection is skipped without performing other checks.
3. If a URL requested by the client belongs to a category of unwanted web resources, the connection is blocked without performing other checks.
4. If the response received from a remote host via HTTP contains a threat belonging to the blocked categories, the connection is blocked without performing other checks.
5. If the data transferred from the local host to a remote host contains a threat belonging to the blocked categories, the connection is blocked without performing other checks.

These five rules are triggered only if `On` is specified in the `InspectHttp` parameter. Otherwise, none of these rules is triggered.

The following six rules specified in `RuleSet9` control the scanning of data sent and received via email protocols (SMTP, POP3 or IMAP); these rules are triggered in the following cases:

- the message contains attachments;
- the message contains a URL belonging to blocked categories;
- the message is qualified as spam with the spam index of no less than 0.8.

If the message is transmitted via the SMTP protocol, an action blocking the transmission (i.e. sending or receipt) of the message is applied, whereas, in case of the IMAP and POP3 protocols, the message is processed such that malicious contents is removed (“repackaging”).

If the Dr.Web Anti-Spam component for scanning of an email message for signs of spam is unavailable, scanning of email messages for signs of spam is not performed. In this case, rules that contain analyzing of a spam level (the `total_spam_score` variable) are unavailable.

The rules for scanning email messages will be triggered only if the corresponding `Inspect`<*EmailProtocol*> parameters are set to `On`. Otherwise, none of these rules are triggered.

The Dr.Web MailD component for email scanning is required for scanning transmitted email messages for malware attachments and signs of spam. If the component is not installed, transmitted email will be blocked because of the error "*Unable to check*". To allow transmitting messages that cannot be checked, set the `BlockUnchecked` parameter to `No` (see above). Moreover, if the email scanning component is not installed, it is recommended to specify `No` for the `InspectSmtp`, `InspectPop3`, and `InspectImap` parameters.

Examples of the rules for traffic monitoring and blocking of access

1. Allow users with IP addresses in the range of *10.10.0.0–10.10.0.254* an access via HTTP to websites of all categories, except *Chats*:

```
protocol in (HTTP), src_ip in (10.10.0.0/24), url_category not in
(Chats) : PASS
```

If the rule:

```
protocol in (HTTP), url_host in "LinuxFirewall.Blacklist" :
BLOCK as BlackList
```

is put on the list of rules above the indicated rule, then access to the domains from the black list, that is, the domains listed in the `LinuxFirewall.Blacklist` parameter, will also be blocked for users with IP addresses in the range of *10.10.0.0–10.10.0.254*. At the same time, if this rule is put below, the users with the addresses in the range of *10.10.0.0–10.10.0.254* will also get access to websites from the black list.

Since the `PASS` resolution is final, no more rules are used; therefore, the downloaded data is not scanned for threats.

To allow users with IP addresses in the range of *10.10.0.0–10.10.0.254* to access websites of all categories except for *Chats*, if they are not on the black list, and to block downloading of threats at the same time, use the following rule:

```
protocol in (HTTP), url_category not in (Chats), url_host not in
"LinuxFirewall.Blacklist", threat_category not in
"LinuxFirewall.BlockCategory" : PASS
```

2. Do not perform scanning of contents of video files *downloaded from the internet* (i.e. data of the `video/*` MIME type, where `*` corresponds to any type of the `video` MIME class):

```
direction response, content_type in ("video/*") : PASS
```

The files uploaded from the local computer (including those that have the `video/*` MIME type) will be scanned because they are sent *in requests, and not in responses*, i.e. the `direction` variable has the `request` value for them.

9.9. Dr.Web File Checker

The Dr.Web File Checker file scanning component is designed for scanning files and directories in the file system. It is used by other components of Dr.Web Security Space to scan file system objects. In addition, this component permanently registers all threats detected in the file system and also functions as a quarantine manager, as it manages the contents of the directories where isolated files are kept.

9.9.1. Operating Principles

The Dr.Web File Checker component is started with superuser (the *root* user) privileges and scans file system objects (files, directories and boot records).

Dr.Web File Checker indexes all scanned files and directories and stores data about scanned objects in a special cache to avoid repeated scanning of the objects that have been already scanned and have not been modified since that (in this case, if a request to scan such an object is received, the previous scan result retrieved from the cache is returned).

When requests to scan file system objects are received from other components, Dr.Web File Checker checks whether the requested object requires scanning. If so, a scanning task is generated for Dr.Web Scanning Engine. If the scanned object contains a threat, Dr.Web File Checker puts it in the registry of detected threats and applies an action to neutralize it (cure, delete or quarantine), if this action has been specified by the client component that initiated the scanning as a reaction to the threat. The scanning can be initiated by various components of Dr.Web Security Space.

During scanning of the requested file system objects, the file-checking component generates a report detailing scanning results and applied actions to neutralize threats, if any, and sends this report to the client component that requested scanning.

Apart from the standard file scanning method, the following special methods are available for internal use:

- *The "flow" method*—a method for scanning files in stream. A component, which uses this method, initializes parameters of scanning and threat neutralization only once. These parameters are further applied to all file scanning requests from this component. This method is used by the SpIDer Guard monitor.
- *The "proxy" method*—a method for file scanning consisting in that the file-checking component only scans files for threats without applying any actions to them and without registering the detected threats (these actions are fully delegated to the component that initiated the scanning). This method is used by the .

Files can be scanned with the "*flow*" method using the `flowscan` command of the Dr.Web Ctl utility (started with the `drweb-ctl` command). However, for a normal on-demand scanning, it is recommended that you use the `scan` command.

During its operation, the file-checking component not only maintains threat registry and manages quarantine, but also collects overall file scan statistics, averaging the number of files scanned per second for the last minute, last 5 minutes, last 15 minutes.

9.9.2. Command-Line Arguments

To start the Dr.Web File Checker component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-filecheck [<parameters>]
```

Dr.Web File Checker accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-filecheck --help
```

This command outputs short help information about the Dr.Web File Checker component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. It is started automatically by the Dr.Web ConfigD configuration management daemon upon receiving requests for file system scanning from other components of Dr.Web Security Space. To manage the operation parameters of the component as well as to scan files on demand, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To scan an arbitrary file or directory using Dr.Web File Checker, you can use the `scan` command of the Dr.Web Ctl tool:

```
$ drweb-ctl scan <path to file or directory>
```

To get documentation for this component from the command line, run the `man 1 drweb-filecheck` command.

9.9.3. Configuration Parameters

The component uses configuration parameters specified in the `[FileCheck]` section of the unified configuration file of Dr.Web Security Space.

This section stores the following parameters:

Parameter	**Description**
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-filecheck`
`DebugClientIpc` *{boolean}*	Log or do not log detailed IPC messages at the debug level (with `LogLevel = DEBUG`). Default value: `No`
`DebugScan` *{boolean}*	Log or do not log detailed messages received during file scanning at the debug level (with `LogLevel = DEBUG`). Default value: `No`
`DebugFlowScan` *{boolean}*	Log or do not log detailed messages about file scanning using the *“flow”* method at the debug level (with `LogLevel = DEBUG`). The *“flow”* method is usually used by the SpIDer Guard monitor. Default value: `No`

Parameter	Description
`DebugProxyScan` *{boolean}*	Log or do not log detailed messages about file scanning using the *“proxy”* method at the debug level (with `LogLevel = DEBUG`). The *“proxy”* method is usually used by . Default value: `No`
`DebugCache` *{boolean}*	Log or do not log detailed messages about the cache status of scanned files at the debug level (with `LogLevel = DEBUG`). Default value: `No`
`MaxCacheSize` *{size}*	Maximum allowed size of cache to store data about scanned files. If `0` is specified, caching is disabled. Default value: `50mb`
`RescanInterval` *{time interval}*	Period of time during which a file will not be rescanned if the results of its previous scan are available in the cache (the period during which the stored information is considered up-to-date). Allowed values: from 0 seconds (`0s`) to 1 minute (`1m`). If the set interval is less than `1s`, there will be no delay—the file will be scanned upon any request. Default value: `1s`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`

9.10. Dr.Web Network Checker

The Dr.Web Network Checker component is designed to scan data received over the network, with the scanning engine, as well as for distributed file scanning for threats. The component allows to establish a connection between network hosts with Dr.Web Security Space installed on them for receiving and sending data (for example, file contents) via the network hosts for scanning. The component manages automatic distribution of scanning tasks (by sending and receiving them over the network) to all available network hosts with which the connection is configured. The component balances the load caused by the scanning tasks between the hosts. If there are no configured connections with remote hosts, the component sends all the data only to the local instance of Dr.Web Scanning Engine.

> This component is always used to scan the data received over network connections. Thus, if the component is absent or unavailable, the operation of the components that send data for scanning over a network connection will be hindered (SpIDer Gate, Dr.Web MailD).
>
> ---
>
> This component is not designed to manage distributed scanning of files located in a local file system, since it cannot replace the Dr.Web File Checker component. To manage distributed scanning of local files, use the Dr.Web MeshD component.
>
> ---
>
> If data scanning over the network is highly intensive, issues with scanning may arise when available file descriptors are depleted. In this case, it is necessary to increase the limit by the number of file descriptors available to Dr.Web Security Space.

Data can be shared either over an insecure channel or over a secure channel via SSL/TLS. To use a secure connection it is required to provide hosts that share files with valid certificates and SSL keys. To generate keys and certificates, you can use the `openssl` utility. An example of how to use the `openssl` utility to generate certificates and private keys is given in the Appendix E. Generating SSL Certificates section.

9.10.1. Operating Principles

The component allows sending the data not represented by files of a local file system for scanning to Dr.Web Scanning Engine located on a local or remote host. This data is processed by the components that send data for scanning over a network connection (SpIDer Gate, Dr.Web MailD).

> These components always use Dr.Web Network Checker (even if it is located on a local host) to send files for scanning to Dr.Web Scanning Engine. Thus, if Dr.Web Network Checker is unavailable, these components *cannot operate correctly*.

In addition, Dr.Web Network Checker allows to connect Dr.Web Security Space to a given set of hosts on the network that run Dr.Web Security Space (or any other Dr.Web for UNIX solution 10.1 or later) in order to manage a distributed search for data not represented by files of the local file system. Thereby, this component allows to create and configure a *scanning cluster*, which is a set of network hosts that exchange data for scanning (each host must run its own instance of the Dr.Web Network Checker distributed scanning agent). On each network host comprised by the scanning cluster, Dr.Web Network Checker performs automatic distribution of tasks for scanning data, sending them over the network to all available hosts for which the connection is configured. At the same time, the host load balancing caused by data scanning is performed depending on the amount of resources available on remote hosts. The number of child scanning processes generated by Dr.Web Scanning Engine on a host comprised by the cluster indicates the amount of resources available for load. The lengths of the queues of the files for scanning on each host in use are also considered.

In this case, any network host comprised by the scanning cluster can act both as a scanning client that sends data for remote scanning and as a scanning server that receives data from the specified network hosts for scanning. If necessary, the distributed scanning agent can be configured so that the host acts only as a scanning server or only as a scanning client.

The data received over the network for scanning is stored on the local file system as temporary files and sent to Dr.Web Scanning Engine or, in case it is unavailable or heavily loaded, to another host of the scanning cluster.

The `InternalOnly` parameter of the component settings allows to manage the Dr.Web Network Checker operation mode. The parameter defines if the component is used for including Dr.Web Security Space in the scanning cluster or only for internal purposes of the Dr.Web Security Space components operating locally.

> You can create your own component (external application) which will use Dr.Web Network Checker to scan files. For this, the Dr.Web Network Checker component provides a custom API based on the Google Protobuf technology. The description of the Dr.Web Network Checker API , as well as client application sample code that uses Dr.Web Network Checker, are supplied as part of the `drweb-netcheck` package.

9.10.2. Command-Line Arguments

To start the Dr.Web Network Checker component from the command line, run the command:

```
$ <opt_dir>/bin/drweb-netcheck [<parameters>]
```

Dr.Web Network Checker accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.

`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-netcheck --help
```

This command outputs short help information about the Dr.Web Network Checker component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary (usually, at the startup of the operating system). At the same time, if a `FixedSocket` parameter value is specified in the component settings and the `InternalOnly` parameter is set to `No`, the component will be started by the configuration management daemon and always available to clients via a UNIX socket. To manage component operation parameters and to start network scanning (if there is a configured connection to other network hosts), use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line. If there is no configured connection to other network hosts, normal scanning will be started using the local scanning engine instead of network scanning.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To scan an arbitrary file or directory with Dr.Web Network Checker, use the `netscan` command of the Dr.Web Ctl tool:

```
$ drweb-ctl netscan <path to file or directory>
```

To get documentation for this component from the command line, run the `man 1 drweb-netcheck` command.

9.10.3. Configuration Parameters

The component uses configuration parameters specified in the `[NetCheck]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
LogLevel *{logging level}*	Logging level of the component. If a parameter value is not specified, the DefaultLogLevel parameter value from the [Root] section is used. Default value: Notice
Log *{log type}*	Logging method of the component. Default value: Auto
ExePath *{path to file}*	Component executable path. Default value: /opt/drweb.com/bin/drweb-netcheck
FixedSocket *{path to file | address}*	Socket of the Dr.Web Network Checker agent fixed instance. If this parameter is specified, the Dr.Web ConfigD configuration management daemon ensures that there is always a running instance of the distributed scanning agent available to clients via this socket. Allowed values: • <*path to file*>—path to a local UNIX socket; • <*address*>—network socket set as an <*IP address*>:<*port*> pair. Default value: *(not specified)*
InternalOnly *{boolean}*	Component operation mode. If the value is set to Yes, the component is used for internal purposes of Dr.Web Security Space components only and is not used for participating in a scanning cluster and for servicing external (to Dr.Web Security Space) client applications regardless of LoadBalance* settings and the specified value of the FixedSocket parameter. Default value: No
RunAsUser *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the "name:" prefix, for example: RunAsUser = name:123456. If the user name is not specified, the component shuts down with an error upon startup. Default value: drweb
IdleTimeLimit *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down.

Parameter	Description
	If the `LoadBalanceAllowFrom` or `FixedSocket` parameter is set, this setting is ignored (the component does not finish its operation after the time interval expires). Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`LoadBalanceUseSsl` *{boolean}*	Use or do not use SSL/TLS to connect to other hosts. Allowed values: • `Yes`—use SSL/TLS; • `No`—do not use SSL/TLS. If the parameter is set to `Yes`, a certificate and a private key must be specified for this host and for all hosts with which it interacts (the `LoadBalanceSslCertificate` and `LoadBalanceSslKey` parameters). Default value: `No`
`LoadBalanceSslCertificate` *{path to file}*	Path to the SSL certificate used by Dr.Web Network Checker on the current host for communication with other hosts via a secure SSL/TLS connection. ⚠ The certificate file and the private key file (specified by the `LoadBalanceSslKey` parameter) must match each other. Default value: *(not specified)*
`LoadBalanceSslKey` *{path to file}*	Path to the private key file used by Dr.Web Network Checker on the current host for communication with other hosts via a secure SSL/TLS connection. ⚠ The certificate file (specified by the `LoadBalanceSslCertificate` parameter) and the private key file must match each other. Default value: *(not specified)*
`LoadBalanceSslCa` *{path}*	Path to the directory or file with the list of trusted root certificates. Among these certificates, there must be a certificate that certifies the authenticity of the certificates used by agents within the scanning cluster when exchanging data via SSL/TLS protocols. If the parameter value is empty, Dr.Web Network Checker operating on the current host does not authenticate certificates of

Parameter	Description
	interacting agents; however, depending on the settings, these agents can verify authenticity of the certificate used by the agent operating on this host. Default value: *(not specified)*
`LoadBalanceSslCrl` *{path}*	Path to the directory or file with a list of revoked certificates. If a parameter value is not specified, Dr.Web Network Checker running on the current host does not verify certificates of interacting agents; however, depending on the settings, they may verify relevance of the certificate used by the agent running on the current host. Default value: *(not specified)*
`LoadBalanceServerSocket` *{address}*	Network socket (an IP address and a port) listened by Dr.Web Network Checker on the current host for receiving files sent by remote hosts for scanning (in case of operating as a network scanning server). Default value: *(not specified)*

`LoadBalanceAllowFrom`
{IP address}
	IP address of a remote network host from which the Dr.Web Network Checker operating on the current host can receive files for scanning (as a network scanning server).
	Accepts a list of values. The values in the list must be comma-separated (with each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add host addresses 192.168.0.1 and 10.20.30.45 to the list.
	1. Adding values to the configuration file.
	• Two values per line:
	`[NetCheck]`
	`LoadBalanceAllowFrom = "192.168.0.1", "10.20.30.45"`
	• Two lines (one value per line):
	`[NetCheck]`
	`LoadBalanceAllowFrom = 192.168.0.1`
	`LoadBalanceAllowFrom = 10.20.30.45`

Parameter	Description

	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset NetCheck.LoadBalanceAllowFrom -a 192.168.0.1`
	`# drweb-ctl cfset NetCheck.LoadBalanceAllowFrom -a 10.20.30.45`
	If the parameter is empty, remote files are not accepted for scanning (the host does not operate as a scanning server).
	Default value: *(not specified)*

`LoadBalanceSourceAddress` *{IP address}*	IP address of a network interface used by Dr.Web Network Checker on the current host to transfer files for remote scanning (if the host operates as a network scanning client and has several network interfaces). If an empty value is specified, the network interface is automatically selected by the OS kernel. Default value: *(not specified)*

`LoadBalanceTo`
{address}
	Socket (an IP address and a port) of a remote host to which Dr.Web Network Checker operating on the current host can send files for remote scanning (as a network scanning client).
	Accepts a list of values. The values in the list must be comma-separated (with each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add sockets 192.168.0.1:1234 and 10.20.30.45:5678 to the list.
	1. Adding values to the configuration file.
	• Two values per line:
	`[NetCheck]`
	`LoadBalanceTo = "192.168.0.1:1234", "10.20.30.45:5678"`
	• Two lines (one value per line):
	`[NetCheck]`
	`LoadBalanceTo = 192.168.0.1:1234`
	`LoadBalanceTo = 10.20.30.45:5678`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset NetCheck.LoadBalanceTo -a 192.168.0.1:1234`
	`# drweb-ctl cfset NetCheck.LoadBalanceTo -a 10.20.30.45:5678`

Parameter	Description
	If the parameter value is empty, local files cannot be transferred for a remote scanning (the host does not operate as a network scanning client). Default value: *(not specified)*
`LoadBalanceStatusInterval` *{time interval}*	Time interval the current host waits to inform all distributed scanning agents specified in the `LoadBalanceAllowFrom` parameter about its workload. Default value: `1s`
`SpoolDir` *{path to directory}*	Local file system directory used to store files received from clients by Dr.Web Network Checker over the network for scanning. Default value: `/tmp/netcheck`
`LocalScanPreference` *{fractional number}*	Relative weight (priority) of the host upon selecting a server to scan a file (a local file or a file received over the network). If at some instant a relative weight of the local host is greater than the total weight of all hosts available as scanning servers, the file is kept by the agent for local scanning. Minimal value: `1`. Default value: `1`

9.11. Dr.Web Scanning Engine

Dr.Web Scanning Engine is designed to search for viruses and other malicious objects in files and boot records (*MBR—Master Boot Record*, *VBR—Volume Boot Record*) of disk devices. The component loads Dr.Web Virus-Finding Engine into memory and starts it as well as loads Dr.Web virus databases used by the engine to detect threats.

The scanning engine operates in daemon mode, as a service that receives requests for scanning file system objects from other Dr.Web Security Space components (these are Dr.Web File Checker and Dr.Web Network Checker and, potentially, Dr.Web MeshD). If Dr.Web Scanning Engine and Dr.Web Virus-Finding Engine are absent or unavailable, no antivirus scanning is performed on this host (except when Dr.Web Security Space contains the Dr.Web MeshD component, whose settings define a connection to local cloud hosts providing scanning engine services).

9.11.1. Operating Principles

The Dr.Web Scanning Engine component operating in daemon mode receives requests from other Dr.Web Security Space components to scan file system objects (files and boot records) for embedded threats, queues scanning tasks and scans requested objects with Dr.Web Virus-Finding Engine. If a threat is detected in a scanned object that must be cured according to the scanning task, the scanning engine attempts to cure it if this action is applicable.

The scanning engine, Dr.Web Virus-Finding Engine and virus databases form a single entity and cannot be separated. The scanning engine downloads the virus databases and provides an operation environment for cross-platform Dr.Web Virus-Finding Engine. The virus databases and the scanning engine are updated by the Dr.Web Updater component included in Dr.Web Security Space but not being a part of the scanning engine. The update component is run by the Dr.Web ConfigD configuration management daemon periodically or forcibly in response to a user command. In addition, if Dr.Web Security Space operates in centralized protection mode, the virus databases and the scanning engine are updated by Dr.Web ES Agent, which interacts with a centralized protection server and receives updates from it.

Dr.Web Scanning Engine can be controlled by the Dr.Web ConfigD configuration management daemon or operate in standalone mode. In the former case, the daemon starts the engine and ensures that the virus databases are up to date. In the latter case, the engine is started and the virus databases are updated by an external application that uses the engine. Both the Dr.Web Security Space components that make requests to the scanning engine for scanning files and external applications use the same API.

> You can create your own component (an external application) using Dr.Web Scanning Engine for scanning files. For this purpose, Dr.Web Scanning Engine provides a custom API based on the Google Protobuf technology. To obtain the Dr.Web Scanning Engine API guide and examples of client application code using Dr.Web Scanning Engine, contact the partner relations department of the Doctor Web company (https://partners.drweb.com/).

Received scanning tasks are automatically distributed in queues with different priorities (high, normal and low). Selection of a queue for a task depends on the component that created the task. For example, tasks received from file system monitors are placed in a high-priority queue, because response time is important while monitoring file system objects. The scanning engine collects statistics on its usage, including the number of all tasks received for scanning and queue lengths. As an average load rate, the scanning engine uses an average length of queues per second. This rate is averaged for the last minute, last 5 minutes and last 15 minutes.

Dr.Web Virus-Finding Engine supports a signature analysis (signature-based detection of threats covered by virus databases) and other methods of heuristic and behavioral analyses designed for detection of potentially dangerous objects based on machine instructions and other attributes of executable code.

> ⚠ Heuristic analysis cannot guarantee highly reliable results and may allow for the following errors:
>
> - *Errors of the first type*. These errors occur when a safe object is detected as malicious (false positive detections).
> - *Errors of the second type*. These errors occur when a malicious object is detected as safe.
>
> Thus, objects detected by the heuristic analyzer are treated as *Suspicious*.

It is recommended that you quarantine suspicious objects. After virus databases are updated, such files can be scanned using the signature analysis. Keep the virus databases up to date in order to avoid errors of the second type.

Dr.Web Virus-Finding Engine allows to scan and cure both unpacked and packed files and objects inside various containers, such as archives, email messages and so on.

9.11.2. Command-Line Arguments

To start Dr.Web Scanning Engine from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-se <socket> [<parameters>]
```

where the mandatory *<socket>* argument indicates an address of a socket used by Dr.Web Scanning Engine for processing requests of client components. It can be set only as a file path (a UNIX socket).

Dr.Web Scanning Engine accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.

`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.
Startup options (identical to configuration file parameters and substitute them when necessary):	
`--CoreEnginePath`	Function: Path to the library of Dr.Web Virus-Finding Engine. Short form: None. Arguments: <*path to file*>—full path to the library in use.
`--VirusBaseDir`	Function: Path to a directory with virus database files. Short form: None. Arguments: <*path to directory*>—full path to the virus database directory.
`--TempDir`	Function: Path to a directory with temporary files. Short form: None. Arguments: <*path to directory*>—full path to the directory with temporary files.
`--KeyPath`	Function: Path to the key file in use. Short form: None. Arguments: <*path to file*>—full path to the key file.
`--MaxForks`	Function: Maximum allowed number of child processes that can be spawned by Dr.Web Scanning Engine during scanning. Short form: None. Arguments: <*number*>—maximum allowed number of child processes.
`--WatchdogInterval`	Description: Frequency with which Dr.Web Scanning Engine checks whether child processes are operable and ends those that stopped responding. Short form: None. Arguments: <*time interval*>—frequency of checking child processes.
`--AbortForkOnTimeout`	Function: Terminate scan processes by sending them SIGABRT on timeout. Short form: None. Arguments: None.
`--ShellTrace`	Function: Shell tracing (log detailed information about file scanning performed by Dr.Web Virus-Finding Engine). Short form: None. Arguments: None.
`--LogLevel`	Description: Logging level of Dr.Web Scanning Engine during its operation. Short form: None. Arguments: <*logging level*>. Allowed values: • `DEBUG`—the most detailed logging level. All messages and debug information are logged. • `INFO`—all messages are logged. • `NOTICE`—all error messages, warnings and notifications are logged. • `WARNING`—all error messages and warnings are logged.

	• ERROR—only error messages are logged.
--Log	Description: Logging method of the component. Short form: None. Arguments: <*log type*>. Allowed values: • Stderr[:ShowTimestamp]—output messages to the standard error stream *stderr*. The ShowTimestamp option is used to add a timestamp to every message. • Syslog[:<*facility*>]—pass messages to the syslog system logging service. The <*facility*> additional option is used to specify a type of a log to store messages logged by syslog. Allowed values: o DAEMON—messages from daemons; o USER—messages from user processes; o MAIL—messages from mailers; o LOCAL0—messages from local processes 0; ... o LOCAL7—messages from local processes 7. • <*path*>—path to a file for storing log messages. Examples: --Log /var/opt/drweb.com/log/se.log --Log Stderr:ShowTimestamp --Log Syslog:DAEMON

Example:

```
$ /opt/drweb.com/bin/drweb-se /tmp/drweb.ipc/.se --MaxForks=5
```

This command starts an instance of Dr.Web Scanning Engine and instructs it to create the /tmp/drweb.ipc/.se UNIX socket to interact with client components and limit the number of child scanning processes to 5.

Startup Notes

If necessary, any number of instances of Dr.Web Scanning Engine can be started, which provide the scanning service for client applications (not only for Dr.Web Security Space components). At that, if a value of the FixedSocketPath parameter is specified in the component settings, one instance of the scanning engine is always run by the Dr.Web ConfigD configuration management daemon and available to the clients via the UNIX socket. Instances of the scanning engine started directly from the command line will operate in standalone mode without establishing connection to the configuration management daemon, even if it is running. To manage the operation of the component, as well as to scan files upon request, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line .

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To scan an arbitrary file or directory with Dr.Web Scanning Engine, use the `rawscan` command of the Dr.Web Ctl tool:

```
$ drweb-ctl rawscan <path to file or directory>
```

To get documentation for this component from the command line, run the `man 1 drweb-se` command.

9.11.3. Configuration Parameters

The component uses configuration parameters specified in the `[ScanEngine]` section of the unified configuration file of Dr.Web Security Space.

This section stores the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-se`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. If the `FixedSocketPath` parameter is set, this setting is ignored (the component does not finish its operation after the time interval expires). Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `1h`
`FixedSocketPath` *{path to file}*	Path to the UNIX socket of the fixed instance of Dr.Web Scanning Engine.

Parameter	Description
	If this parameter is specified, the Dr.Web ConfigD configuration management daemon ensures that there is always a running scanning engine instance available to clients via this socket. Default value: *(not specified)*
`MaxForks` *{integer}*	Maximum allowed number of child scanning processes that can run simultaneously spawned by Dr.Web Scanning Engine. Default value: *Automatically determined at startup* as twice the number of available CPU cores; or `4`, if the resulting number is less than 4.
`WatchdogInterval` *{time interval}*	Rate at which Dr.Web Scanning Engine checks whether child scanning processes spawned by it are operable to detect deadlocks (“watchdog timer”). Default value: `15s`
`BufferedIo` *{boolean}*	Enable or disable buffered input/output while scanning files. Using buffered input/output on OSes of the GNU/Linux family can increase speed of scanning files on slow disks. Allowed values: • `On`, `Yes`, `True`—enable buffered input/output; • `Off`, `No`, `False`—disable buffered input/output. Default value: `Off`
`AbortForkOnTimeout` *{boolean}*	Terminate or do not terminate scan processes by sending them SIGABRT on timeout. Allowed values: • `On`, `Yes`, `True`—terminate scan processes on timeout; • `Off`, `No`, `False`—do not terminate scan processes on timeout. Default value: `Off`

9.12. Dr.Web Updater

The Dr.Web Updater component is designed for receiving all available updates for virus databases and Dr.Web Virus-Finding Engine from Doctor Web update servers and synchronize updates with a local cloud of Dr.Web for UNIX products via the Dr.Web MeshD component.

If Dr.Web Security Space operates in centralized protection mode, updates are received from a centralized protection server; at that, all updates are received from the server via Dr.Web ES Agent, and Dr.Web Updater is not used for downloading updates (the local cloud of Dr.Web for UNIX products is also not used to synchronize updates).

9.12.1. Operating Principles

The component is designed to establish connections to Doctor Web update servers to check for updates for virus databases and Dr.Web Virus-Finding Engine, for the database of web resource categories and for the anti-spam component. The lists of the servers that constitute an available update zone are stored in a special file signed to prevent its modification. Only basic and digest authentication are supported for connection to the update servers using a proxy server.

If Dr.Web Security Space is not connected to a centralized protection server or is connected to it in mobile mode, Dr.Web Updater is automatically started by the Dr.Web ConfigD configuration management daemon at intervals specified in the settings. The component can also be started by Dr.Web ConfigD upon receiving a corresponding command from the user (unscheduled update).

When updates become available on update servers, they are downloaded to the `/var/opt/drweb.com/cache/`; after that they are moved to the working directories of Dr.Web Security Space.

By default, all updates are downloaded from the update zone that is common for all Dr.Web products. The list of the servers used by default and included in the update zone is specified in the files located in directories defined by `*DrlDir` parameters grouped by an update type: for virus databases and the scanning engine, for the database of web resource categories, for the antispam component). Upon a client request a custom update zone can be created (for each update type), the server list of which is specified in a separate file (named `custom.drl` by default) located in the directory specified by the corresponding `*CustomDrlDir` parameter. In this case, the update component will download updates only from these servers without using the servers from the default zone.

If you do not want to use the custom update zone, clear the value of the corresponding `*CustomDrlDir` parameter in the component settings.

> The content of the files with server lists is signed so that the files cannot be modified. If you need to create a custom list of update servers, contact our technical support.

The component can back up the files to be updated to further roll back updates at user request. You can specify a backup location and an update history depth in the component settings. To roll back updates, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line (run the tool with the `drweb-ctl` command).

If Dr.Web Security Space is connected to a local cloud of Dr.Web for UNIX products and not connected to a centralized protection server, the Dr.Web Updater component is used to synchronize updates received by cloud hosts as well, that is, it sends latest updates received from update servers to the cloud and receives latest updates from the cloud, which allows to reduce the total load of Dr.Web update servers. This option can be enabled or disabled in the component settings.

9.12.2. Command-Line Arguments

To start the Dr.Web Updater component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-update [<parameters>]
```

Dr.Web Updater accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-update --help
```

This command outputs short help information about the Dr.Web Updater component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration daemon when necessary. To manage the operation parameters of the component, as well as to update virus databases and the scan engine, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To get documentation for this component from the command line, run the `man 1 drweb-update` command.

9.12.3. Configuration Parameters

The component uses configuration parameters specified in the `[Update]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-update`
`RunAsUser` *{UID | user name}*	The parameter determines on behalf of which user the component runs. Either the numerical UID of the user or the user name (login) can be specified. If the user name consists of numbers (i.e. similar to a numerical UID), it must be specified with the "`name:`" prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`UpdateInterval` *{time interval}*	The frequency to check for updates on Dr.Web update servers. This is a time period between a previous successful attempt to connect to the update servers (initiated automatically or manually) and the next attempt to perform an update. Default value: `30m`
`RetryInterval` *{time interval}*	Frequency of repeated attempts to perform an update using the update servers if the previous attempt failed. Allowed values: from 1 minute (`1m`) to 30 minutes (`30m`). Default value: `3m`

Parameter	Description
`MaxRetries` *{integer}*	Number of repeated attempts to perform an update using Dr.Web update servers (the frequency is specified in the `RetryInterval` parameter) if the previous attempt failed. If the value is set to `0`, repeated attempts are not made (the next update will be performed after the time period specified in the `UpdateInterval` parameter). Default value: `3`

`Proxy`
{connection string}
	Parameters for connecting to a proxy server that is used by the Dr.Web Updater component when it is connecting to Dr.Web update servers (for example, if connecting directly to external servers is prohibited by network security policies).
	If a parameter value is not specified, the proxy server is not used.
	Allowed values:
	<connection string>—proxy server connection string. The string has the following format (URL):
	`[`*<protocol>*`://][`*<user>*`:`*<password>*`@]`*<host>*`:`*<port>*
	where:
	• *<protocol>* is a type of the protocol in use (in the current version, only `http` is available);
	• *<user>* is a user name for connecting to the proxy server;
	• *<password>* is a password for connecting to the proxy server;
	• *<host>* is the host address of the proxy server (an IP address or a domain name, i.e. FQDN);
	• *<port>* is a port to be used.
	The URL parts *<protocol>* and *<user>:<password>* may be absent. The proxy server address *<host>:<port>* is mandatory.
	If the user name or the password contains characters `@`, `%` or `:`, these characters must be replaced with the corresponding HEX codes: `%40`, `%25` and `%3A` respectively.
	Examples:
	1. In the configuration file:
	• Connection to a proxy server hosted at `proxyhost.company.org` using port `123`:
	`Proxy = proxyhost.company.org:123`
	• Connection to a proxy server hosted at `10.26.127.0` using port `3336` via HTTP protocol as the `legaluser` user with the `passw` password:
	`Proxy = http://legaluser:passw@10.26.127.0:3336`
	• Connection to the proxy server hosted at `10.26.127.0` using port `3336` as the `user@company.com` user with the `passw%123` password:
	`Proxy = user%40company.com:passw%25123%3A@10.26.127.0:3336`

Parameter	Description

	2. Setting the same values using the `drweb-ctl cfset` command:
	`# drweb-ctl cfset Update.Proxy proxyhost.company.org:123`
	`# drweb-ctl cfset Update.Proxy http://legaluser:passw@10.26.127.0:3336`
	`# drweb-ctl cfset Update.Proxy user%40company.com:passw%25123%3A@10.26.127.0:3336`
	Default value: *(not specified)*

`ExcludedFiles`
{file name}
	Name of a file that will not be updated by the Dr.Web Updater component.
	Accepts a list of values. The values in the list must be comma-separated (with each value put in quotation marks). The parameter can be specified more than once in the section (in this case, all its values are combined into one list).
	Example: Add the `123.vdb` and `456.dws` files to the list.
	1. Adding values to the configuration file.
	• Two values per string:
	`[Update]`
	`ExcludedFiles = "123.vdb", "456.dws"`
	• Two strings (one value per string):
	`[Update]`
	`ExcludedFiles = 123.vdb`
	`ExcludedFiles = 456.dws`
	2. Adding values with the `drweb-ctl cfset` command:
	`# drweb-ctl cfset Update.ExcludedFiles -a 123.vdb`
	`# drweb-ctl cfset Update.ExcludedFiles -a 456.dws`
	Default value: `drweb32.lst`

`NetworkTimeout` *{time interval}*	A time-out period imposed on the network-related operations of the component while downloading updates from Dr.Web servers. This parameter is used when a connection is temporarily lost: if the connection is established again before the time-out expires, the interrupted updating process will be continued. Specifying the time-out value larger than `75s` has no effect as the connection is closed by the TCP timeout. Minimal value: `5s`. Default value: `60s`
`BaseDrlDir` *{path to directory}*	Path to a directory that contains files used for connection to servers of a standard update zone, which are used to update virus databases and the scan engine.

Parameter	Description
	Default value: `/var/opt/drweb.com/drl/bases`
`BaseCustomDrlDir` *{path to directory}*	Path to a directory that contains files used for connection to a special ("customized") update zone, which are used to update virus databases and the scan engine. If the directory defined in this parameter contains a non-empty signed server list file (the `.drl` file), the update is performed only from these servers, and the main zone servers (see above) are not used to update the virus databases and the scanning engine. Default value: `/var/opt/drweb.com/custom-drl/bases`
`BaseUpdateEnabled` *{boolean}*	Allow or do not allow updating the virus databases and the scan engine. Allowed values: • `Yes`—updating is allowed and will be performed; • `No`—updating is not allowed and will not be performed. Default value: `Yes`
`VersionDrlDir` *{path to directory}*	Path to a directory that contains files for connection to servers that are used for updating versions of Dr.Web Security Space components. Default value: `/var/opt/drweb.com/drl/version`
`VersionUpdateEnabled` *{boolean}*	Allow or do not allow updating versions of Dr.Web Security Space components. Allowed values: • `Yes`—updating is allowed and will be performed; • `No`—updating is not allowed and will not be performed. Default value: `Yes`
`DwsDrlDir` *{path to directory}*	Path to a directory that contains the files for connecting to servers of a standard update zone, which are used for updating the database of web resource categories. Default value: `/var/opt/drweb.com/drl/dws`
`DwsCustomDrlDir` *{path to directory}*	Path to a directory that contains the files for connecting to servers of a special ("customer") update zone, which are used for updating the database of web resource categories. If the directory defined in this parameter contains a non-empty signed server list file (the `.drl` file), updating is performed only from these servers, and the main zone servers (see above) are not used to update the database of web resource categories. Default value: `/var/opt/drweb.com/custom-drl/dws`
`DwsUpdateEnabled` *{boolean}*	Allow or do not allow updating of the database of web resource categories. Allowed values: • `Yes`—updating is allowed and will be performed;

Parameter	Description
	• `No`—updating is not allowed and will not be performed. Default value: `Yes`
`AntispamDrlDir` *{path to directory}*	Path to a directory that contains the files for connecting to servers of a standard update zone, which are used for updating the anti-spam library. Default value: `/var/opt/drweb.com/drl/antispam`
`AntispamCustomDrlDir` *{path to directory}*	Path to a directory that contains the files for connecting to servers of a special ("customer") update zone, which are used for updating the anti-spam library. If the directory defined in this parameter contains a non-empty signed server list file (the `.drl` file), updating is performed only from these servers, and the main zone servers (see above) are not used to update the anti-spam library. Default value: `/var/opt/drweb.com/custom-drl/antispam`
`AntispamUpdateEnabled` *{boolean}*	Allow or do not allow updating of the anti-spam library. Allowed values: • `Yes`—updating is allowed and will be performed; • `No`—updating is not allowed and will not be performed. Default value: `No`
`BackupDir` *{path to directory}*	Path to a directory where old versions of updated files are saved for possible rollback. Only updated files are backed up upon every update. Default value: `/tmp/update-backup`
`MaxBackups` *{integer}*	The maximum number of the previous versions of updated files, which are saved. If this number is exceeded, the oldest copy is removed upon the next update. If the parameter value is `0`, the previous versions of backup files are not stored. Default value: `0`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. The component is started upon the next update on schedule. When the update is completed, it is waiting for the specified time interval, and, if there are no new requests, it shuts down until the next update attempt. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`Start` *{boolean}*	Enable or disable the component at the startup of Dr.Web Security Space. This parameter has priority over the `DwsUpdateEnabled` parameter.

Parameter	Description
	Allowed values: • `Yes`—enable the component at the startup of Dr.Web Security Space; • `No`—disable the component at the startup of Dr.Web Security Space. Default value: `Yes`
`UseHttps` *{Always | ResListOnly | Never}*	Use or do not use HTTPS while downloading updates. Allowed values: • `Always`—always use HTTPS while downloading updates. • `ResListOnly`—use HTTPS only while downloading an `.lst` file containing a list of update files. At that, update files will be downloaded via HTTP. • `Never`—always use HTTP while downloading updates. Default value: `ResListOnly`

9.13. Dr.Web ES Agent

The centralized protection agent Dr.Web ES Agent is designed for connecting Dr.Web Security Space to a centralized protection server.

When Dr.Web Security Space is connected to the centralized protection server, Dr.Web ES Agent synchronizes the license key file with the key files stored on the centralized protection server. In addition, Dr.Web ES Agent sends statistics on virus events, the list of running components and their status to the centralized protection server.

Furthermore, Dr.Web ES Agent updates Dr.Web Security Space virus databases directly from the connected centralized protection server without using the Dr.Web Updater component.

9.13.1. Operating Principles

The Dr.Web ES Agent component connects to a centralized protection server, which allows a network administrator to implement a common security policy within the entire network, in particular, to configure the same scanning settings and reaction on threat detection for all workstations and servers of the network. In addition, the centralized protection server also acts as an internal update server on the protected network and stores up-to-date virus databases (in this case, updating is performed via Dr.Web ES Agent, Dr.Web Updater is not used).

When Dr.Web ES Agent connects to the centralized protection server, the agent receives up-to-date settings of the program components and the license key file from the server, which are then passed to the Dr.Web ConfigD configuration management daemon for applying them to the managed components. In addition, the component can also receive tasks from the centralized protection server for scanning file system objects on the station (including on schedule).

Dr.Web ES Agent collects statistics on detected threats and applied actions and sends it to the server to which the agent is connected.

To connect Dr.Web ES Agent to the centralized protection server, the password and identifier of the host (“station” in terms of the centralized protection server) are required as well as a public encryption key file, which is used by the server for authentication. Instead of the station identifier, while connecting, you can specify the identifier of the main and tariff groups in which the station is to be included. For the required identifiers and public key file, contact the administrator of your anti-virus network.

In addition, you can connect a protected host (“station”) to the centralized protection server as a “newbie”, if this is allowed by the centralized protection server. In this case, after the administrator confirms the request to connect the station, the centralized protection server automatically generates new identifier and password for the host and sends them to the agent for future connections.

9.13.2. Command-Line Arguments

To start the Dr.Web ES Agent component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-esagent [<parameters>]
```

Dr.Web ES Agent accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-esagent --help
```

This command outputs short help information about the Dr.Web ES Agent component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration daemon at the startup of the operating system. To manage the operation parameters of the component as well as to connect Dr.Web Security Space to a centralized protection server, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To get documentation for this component from the command line, run the `man 1 drweb-esagent` command.

9.13.3. Configuration Parameters

The component uses configuration parameters specified in the `[ESAgent]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-esagent`
`DebugIpc` *{logical}*	Log or do not log IPC messages at the debug level (with `LogLevel = DEBUG`) (interaction of Dr.Web ES Agent with the Dr.Web ConfigD configuration management daemon). Default value: `No`
`MobileMode` *{On | Off | Auto}*	Enable or disable the mobile mode of Dr.Web Security Space while connected to a centralized protection server. Allowed values: • `On`—enable the mobile mode if it is allowed by the centralized protection server (install updates from the update servers of Doctor Web using Dr.Web Updater); • `Off`—disable the mobile mode and continue operation in centralized protection mode (always receive updates from the centralized protection server); • `Auto`—enable the mobile mode if allowed by the centralized protection server and install updates both from the update servers of Doctor Web using Dr.Web Updater and from the centralized protection server depending on which connection is available and which connection quality is higher. Behavior of this parameter depends on server permissions: if the mobile mode is not allowed on the server in use, this parameter has no effect. Default value: `Auto`
`Discovery` *{On | Off}*	Allow or do not allow the agent to receive *discovery* requests from a network inspector built in the centralized protection server (*discovery* requests are used by the inspector to check the structure and state of the anti-virus network).

Parameter	Description
	Allowed values: • `On`—allow the agent to receive and process *discovery* requests; • `Off`—do not allow the agent to receive and process *discovery* requests. This parameter has priority over the settings of the centralized protection server: if the parameter value is set to `Off`, the agent does not receive discovery requests even if this feature is enabled on the server. Default value: `On`
`UpdatePlatform` *{platform name}*	Designation of a platform for which the agent will receive updates for the scanning engine from the centralized protection server. Allowed values: • for GNU/Linux: `unix-linux-32`, `unix-linux-64-engine64`, `unix-linux-aarch64`, `unix-linux-e2k`, `unix-linux-e2k-engine64`, `unix-linux-mips`, `unix-linux-ppc64le`; • for FreeBSD: `unix-freebsd-32`, `unix-freebsd-64-engine64`; • for Darwin: `unix-darwin-32`, `unix-darwin-64-engine64`, `unix-darwin-aarch64`. It is strongly recommended to change the parameter value only if you are sure that this is necessary. Default value: *Depends on the platform in use*
`SrvMsgAutoremove` *{integer}*	Storage period after which the messages from the centralized protection server are removed automatically. Allowed values: from 1 week (`1w`) to 365 days (`365d`). The storage period is specified as an integer with a suffix (`s`, `m`, `h`, `d`, `w`). Default value: `1w`

9.14. Dr.Web MeshD

The Dr.Web MeshD component is an agent that integrates a host on which Dr.Web Security Space is installed with a “local cloud”, which combines hosts with installed Dr.Web for UNIX products. This cloud allows to perform the following tasks:

- providing file scanning services by some cloud hosts to others (a service of providing the scanning engine);
- distributing updates for virus databases among cloud hosts.

To combine hosts with Dr.Web for UNIX products installed, the Dr.Web MeshD component, which integrates a host with the cloud, must be installed on every host. Host privileges within the cloud and cloud features used by a host are flexibly adjusted with Dr.Web MeshD settings.

Data is shared with other cloud hosts over a protected SSH channel.

9.14.1. Operating Principles

In this section

- Connection Types
- Operation Modes
- Services

Dr.Web MeshD is a mediator that ensures interaction of a host with Dr.Web Security Space installed and other cloud hosts.

Connection Types

Dr.Web MeshD uses the following connection types:

- *Client (service)*—used by Dr.Web MeshD to connect to other cloud hosts that are clients of services provided by the given host.

> Dr.Web Security Space components operating on the host and using services provided by the cloud connect to Dr.Web MeshD, which operates on the same host, through a local UNIX socket. At that, a client connection is not used.

- *Partner (peer to peer)*—used by Dr.Web MeshD for interaction with peer (within a service) partner cloud hosts. Usually such horizontal connections are used for scaling and distributing the load when interacting with the cloud, as well as for synchronization of cloud hosts.
- *Uplink*—used by Dr.Web MeshD for connecting this host as a client to cloud hosts that provide services (for example, distribution of virus database updates, sending files for scanning and so on).

The use of all three types of connections is configured for different cloud services independently from each other. At that, the same host can be configured as a server for processing client requests within one service (for example, for distributing latest updates) and as a client within another service (for example, remote file scanning).

Within cloud, hosts perform authorized interaction via SSH, that is, all sides of interhost communication are always mutually authenticated. For the authentication, *host keys* are used in compliance with RFC 4251. A client connection from a local component is always considered as trusted.

Operation Modes

Dr.Web MeshD can either operate in daemon mode or run at the request of other Dr.Web Security Space components installed on the local host. If Dr.Web MeshD is configured to serve client connections (the `ListenAddress` parameter is not empty) and at least one of the services is activated, Dr.Web MeshD starts as a daemon and awaits client connections.

If Dr.Web MeshD is not set to process client connections (the `ListenAddress` parameter is empty) and there are no requests to this component during a time interval specified by the `IdleTimeLimit` parameter, the component shuts down automatically.

Services

Exchanging updates (Update)

This service allows the host to subscribe to updates of virus and other databases, send notifications of the latest updates, upload and distribute the update files among cloud hosts. The service settings can be configured using the `Update*` parameters.

A common service use case assumes that Dr.Web MeshD is installed on a number of machines (clients of the service) in the local network of a company with the feature of obtaining updates enabled. The typical client settings are as follows:

```
[MeshD]
ListenAddress =
```

The following settings are specified on the host acting as a local server for distributing updates:

```
ListenAddress = <address>:<port>
```

Here, <*server address*> in the uplink connection of the client must refer to the <*address*> and <*port*> that are used by the server host for managing client connections.

When one of the hosts is being updated from the update servers (that are external to the local cloud—Dr.Web GUS update servers or a centralized protection server), the host sends a notification to all concerned clients (if the host is configured as a server providing an update exchange service) and sends to the server host a new list of files available for distribution from this host. Upon receiving this notification, client hosts can request downloading updated files from the server, which in turn can request the files from the client to store them locally or to send them to another client that requested these files from the server.

Such mechanism decreases a delay in applying updates because clients send requests to Dr.Web GUS at different times, at that the first updated client immediately distributes the latest update files to all concerned cloud hosts. This also decreases traffic and Dr.Web GUS load.

> When using a local cloud to distribute updates, both the Dr.Web MeshD and Dr.Web Updater components must be installed on hosts.

Remote file scanning (Engine)

This service allows to use Dr.Web Scanning Engine for scanning remote files: hosts acting as clients send files for scanning to a server host, and server hosts provide a service for scanning files sent by the client hosts. Typical client settings are as follows:

```
…
[MeshD]
EngineChannel = On
EngineUplink = <server address>
ListenAddress =
…
```

The following settings are specified on the host acting as a local scanning server:

```
EngineChannel = On
EngineUplink =
ListenAddress = <address>:<port>
```

Here, *<server address>* in the uplink connection of the client must refer to the *<address>* and *<port>* that are used by the server host for managing client connections.

Sending files for scanning (File)

This feature is not used (remote scanning is provided by the *Engine* service).

URL checking (Url)

This service allows to check whether a URL belongs to potentially dangerous and unwanted categories: client hosts send a URL to be checked to a server host. Typical client settings are as follows:

```
…
[MeshD]
UrlChannel = On
UrlUplink = <server address>
ListenAddress =
…
```

The following settings are specified on the host acting as a URL checking server:

```
UrlChannel = On
UrlUplink =
ListenAddress = <address>:<port>
```

Here, *<server address>* in the uplink connection of the client must refer to the *<address>* and *<port>* that are used by the server host for managing client connections.

9.14.2. Command-Line Arguments

To start the Dr.Web MeshD component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-meshd [<parameters>]
```

Dr.Web MeshD accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-meshd --help
```

This command outputs short help information about the Dr.Web MeshD component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

> To start the Dr.Web Ctl tool, run the `drweb-ctl` command.
>
> To get documentation for this component from the command line, run the `man 1 drweb-meshd` command.

9.14.3. Configuration Parameters

The component uses configuration parameters specified in the `[MeshD]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-meshd`
`DebugSsh` *{boolean}*	Log or do not log SSH messages (messages and data are transmitted via SSH) received and sent by the Dr.Web MeshD component operating on the host, if the logging level is `LogLevel = Debug`. Default value: `No`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`

Parameter	Description
`DnsResolverConfPath` *{path to file}*	Path to a DNS resolver configuration file. Default value: `/etc/resolv.conf`
`ListenAddress` *<IP address>:<port>*	Network socket (address and port) of the client connection where the component is waiting to receive connections from cloud hosts. These hosts are clients of services provided by this cloud host. In order for the component to listen on the IPv6 interface and detect cloud client hosts via IPv6, this parameter must be defined. If the value is not specified, the component does not receive requests from clients. Default value: `0.0.0.0:7090` if the `drweb-scanning-server` package is installed, not specified otherwise
`FileChannel` *{On | Off}*	Allow or do not allow the Dr.Web MeshD component operating on this host to participate in file exchange services. If the parameter is set to `On`, the Dr.Web MeshD component is automatically started by the Dr.Web ConfigD configuration daemon. Default value: `On`
`FileUplink` *{address}*	Address of a higher-level host of Dr.Web MeshD receiving files from this host for scanning. Allowed values: • *value is not specified*—server is not specified for this service, and Dr.Web MeshD will not establish connections. • *<IP address>:<port>*—Dr.Web MeshD will connect to a server with the specified address and port. • `dns:`*<service name>*`[:`*<domain>*`]`—address and port of the server are determined by searching for the SRV record of the DNS domain <*domain*>. If <*domain*> is not specified, the domain from the DNS resolver configuration file is used (the path is specified by `ResolverConfPath`). The domain is taken from the `search` and `domain` fields, depending on which of them is encountered last in the configuration file. • `discover`—search for a higher-level host using the *discovery* mechanism. Default value: *(not specified)*
`FileDebugIpc` *{boolean}*	Log or do not log the debug information for the file exchange service if the logging level is `LogLevel = Debug`. Default value: `No`
`EngineChannel` *{On | Off}*	Allow or do not allow the Dr.Web MeshD component operating on this host to provide scanning engine services. If the parameter is set to `On`, the Dr.Web MeshD component is automatically started by the Dr.Web ConfigD configuration daemon.

Parameter	Description
	Default value: `On`
`EngineUplink` *{address}*	Address of a higher-level host of Dr.Web MeshD providing scanning engine services for this host. Allowed values: • *value is not specified*—server is not specified for this service, and Dr.Web MeshD will not establish connections. • <*IP address*>:<*port*>—Dr.Web MeshD will connect to a server with the specified address and port. • `dns:`<*service name*>`[:`<*domain*>`]`—address and port of the server are determined by searching for the SRV record of the DNS domain <*domain*>. If <*domain*> is not specified, the domain from the DNS resolver configuration file is used (the path is specified by `ResolverConfPath`). The domain is taken from the `search` and `domain` fields, depending on which of them is encountered last in the configuration file. • `discover`—search for a higher-level host using the *discovery* mechanism. Default value: *(not specified)*
`EngineDebugIpc` *{boolean}*	Log or do not log the debug information for the file exchange service if the logging level is `LogLevel = Debug`. Default value: `No`
`UrlChannel` *{On | Off}*	Allow or do not allow the Dr.Web MeshD component operating on this host to provide URL checking services. Default value: `On`
`UrlUplink` *{address}*	Address of a higher-level host of Dr.Web MeshD providing URL checking services for this host. Allowed values: • *value is not specified*—URL checking server is not specified. • <*IP address*>:<*port*>—Dr.Web MeshD will connect to a server with the specified address and port. • `dns:`<*service name*>`[:`<*domain*>`]`—address and port of the server are determined by searching for the SRV record of the DNS domain <*domain*>. If <*domain*> is not specified, the domain from the DNS resolver configuration file is used (the path is specified by `ResolverConfPath`). The domain is taken from the `search` and `domain` fields, depending on which of them is encountered last in the configuration file. • `discover`—search for a higher-level host using the *discovery* mechanism. Default value: *(not specified)*
`DiscoveryResponderPort` *{port number}*	Port on which a higher-level MeshD host responds to client requests via UDP. The *discovery* mechanism is enabled only if the `ListenAddress` parameter is defined.

Parameter	Description
	Default value: `18008`
`UrlDebugIpc` *{boolean}*	Log or do not log the debug information for the URL checking service if the logging level is `LogLevel = Debug`. Default value: `No`

> The current version of Dr.Web Security Space does not use the *File* service for sending files for scanning. Use the *Engine* service of the scanning engine instead.

9.15. Dr.Web URL Checker

Dr.Web URL Checker is an auxiliary component for checking whether URLs refer to malicious or unwanted web resources.

Dr.Web URL Checker is used by the following components:

- Dr.Web MeshD,
- SpIDer Gate,
- Dr.Web MailD

9.15.1. Operating Principles

The Dr.Web URL Checker component is designed to check URLs for belonging to unwanted or potentially dangerous categories.

The check can be performed either by using specialized link bases or by using the Dr.Web CloudD service. To use the Dr.Web CloudD service, run the command:

```
# drweb-ctl cfset Root.UseCloud Yes
```

Dr.Web URL Checker cannot be started by the user. This component is started by the Dr.Web ConfigD configuration management daemon upon request of other components.

9.15.2. Command-Line Arguments

To start Dr.Web URL Checker from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-urlcheck [<parameters>]
```

Dr.Web URL Checker accepts the following parameters:

Parameter	Description

`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h`. Arguments: None
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v`. Arguments: None

Example:

```
$ /opt/drweb.com/bin/drweb-urlcheck --help
```

This command outputs short help information about the Dr.Web URL Checker component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

To start the Dr.Web Ctl tool, run the `drweb-ctl` command.

To get documentation for this component from the command line, run the `man 1 drweb-urlcheck` command.

9.15.3. Configuration Parameters

The component uses configuration parameters specified in the `[Urlcheck]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	Description
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component.

Parameter	Description
	Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-urlcheck`
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the "`name:`" prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `None` if the `drweb-scanning-server` package is installed, `10m` otherwise

9.16. Dr.Web CloudD

The Dr.Web CloudD component is designed to communicate with the Dr.Web Cloud service of the Doctor Web company. The Dr.Web Cloud service collects up-to-date information from all Dr.Web anti-virus products about detected threats to prevent users from visiting unwanted websites and protect operating systems of servers and workstations from infected files containing latest threats that are not yet covered by Dr.Web virus databases. Moreover, using Dr.Web Cloud reduces a number of false positives of Dr.Web Scanning Engine.

9.16.1. Operating Principles

The component is designed to communicate with the Dr.Web Cloud service of the Doctor Web company to scan contents of a specified file for threats unknown to local Dr.Web Scanning Engine and to check whether a specified URL belongs to any of the categories of web resources predefined by the Doctor Web company. Furthermore, the component periodically sends the statistics on detection of infected files and information about the operating system, on which Dr.Web Security Space is run, to the Dr.Web Cloud service.

Dr.Web CloudD is automatically run by the configuration daemon upon receiving a command from the user or some component of Dr.Web Security Space.

To use the Dr.Web Cloud service, run the command:

```
# drweb-ctl cfset Root.UseCloud Yes
```

This component is used for communicating with Dr.Web Cloud by the SpIDer Gate component, which monitors network traffic and scans URLs, for scanning of URLs requested by the user.

Furthermore, the component is used while scanning files on demand of the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line (the tool is started with the `drweb-ctl` command). Upon detection of threats, Dr.Web Scanning Engine sends a report about the file to Dr.Web Cloud.

9.16.2. Command-Line Arguments

To start the Dr.Web CloudD component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-cloudd [<parameters>]
```

Dr.Web CloudD accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component.

	Short form: `-h` Arguments: None.
`--version`	Function: Output information about the version of the component to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-cloudd --help
```

This command outputs short help information about the Dr.Web CloudD component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

> To start the Dr.Web Ctl tool, run the `drweb-ctl` command.
>
> ---
>
> To get documentation on this component from the command line, run the `man 1 drweb-cloudd` command.

9.16.3. Configuration Parameters

The Dr.Web CloudD component uses configuration parameters specified in the `[CloudD]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	**Description**
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath`	Component executable path.

Parameter	Description
{path to file}	Default value: `/opt/drweb.com/bin/drweb-cloudd`
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the `name:` prefix, for example, `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down. Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the SIGTERM signal will not be sent if the component goes idle. Default value: `1h`
`PersistentCache` *{On | Off}*	Enable or disable saving of the cache of Dr.Web Cloud responses to the disk. Default value: `Off`
`DebugSdk` *{logical}*	Log or do not log detailed messages from Dr.Web Cloud at the debug level (with `LogLevel = DEBUG`). Default value: `No`

9.17. Dr.Web StatD

The Dr.Web StatD component is designed for accumulating statistics on events that occur during the operation of the Dr.Web Security Space components. The events are registered in permanent storage and can be obtained on request.

9.17.1. Operating Principles

The Dr.Web StatD component ensures accumulation and permanent storage of events obtained during the operation of Dr.Web Security Space components. The following types of events are logged:

- unexpected component shutdown;
- threat detection in an email message.

Dr.Web StatD operates in daemon mode and is automatically started by the Dr.Web ConfigD configuration management daemon. To view and manage events, use the `events` command of the Dr.Web Ctl utility.

9.17.2. Command-Line Arguments

To start the Dr.Web StatD component from the command line, run the following command:

```
$ <opt_dir>/bin/drweb-statd [<parameters>]
```

Dr.Web StatD accepts the following parameters:

Parameter	Description
`--help`	Function: Output short help information about command-line parameters to the console or the terminal emulator and shut down the component. Short form: `-h` Arguments: None.
`--version`	Function: Output information about the component version to the console or the terminal emulator and shut down the component. Short form: `-v` Arguments: None.

Example:

```
$ /opt/drweb.com/bin/drweb-statd --help
```

This command outputs short help information about the Dr.Web StatD component.

Startup Notes

The component cannot be started directly from the command line in standalone mode. The component is started automatically by the Dr.Web ConfigD configuration management daemon when necessary. To manage the operation parameters of the component, use the Dr.Web Ctl tool designed to manage Dr.Web Security Space from the command line.

> To start the Dr.Web Ctl tool, run the `drweb-ctl` command.
>
> ---
>
> To get documentation for this component from the command line, run the `man 1 drweb-statd` command.

9.17.3. Configuration Parameters

The component uses configuration parameters specified in the `[StatD]` section of the unified configuration file of Dr.Web Security Space.

The section contains the following parameters:

Parameter	**Description**
`LogLevel` *{logging level}*	Logging level of the component. If a parameter value is not specified, the `DefaultLogLevel` parameter value from the `[Root]` section is used. Default value: `Notice`
`Log` *{log type}*	Logging method of the component. Default value: `Auto`
`ExePath` *{path to file}*	Component executable path. Default value: `/opt/drweb.com/bin/drweb-statd`
`RunAsUser` *{UID | user name}*	User on behalf of whom the component is started. Either a numerical UID of the user or a user name (login) can be specified. If the user name consists of numbers (that is, the name is similar to a numerical UID), it must be specified with the “`name:`” prefix, for example: `RunAsUser = name:123456`. If the user name is not specified, the component shuts down with an error upon startup. Default value: `drweb`
`IdleTimeLimit` *{time interval}*	Maximum idle time for the component. When the specified period of time expires, the component shuts down.

Parameter	Description
	Allowed values: from 10 seconds (`10s`) to 30 days (`30d`). If the `None` value is set, the component will operate indefinitely; the `SIGTERM` signal will not be sent if the component goes idle. Default value: `10m`
`MaxEventStoreSize` *{size}*	Maximum allowed size of the event database. Defined in `mb`, for example: `MaxEventStoreSize = 100mb`. Minimal value: `50mb`. Default value: `1GB`

10. Appendices

10.1. Appendix A. Types of Computer Threats

Herein, the term *"threat"* is defined as any kind of software potentially or directly capable of inflicting damage to a computer or network and compromising user information or rights (that is, malicious and other unwanted software). Broadly speaking, the term "threat" can be used to indicate any type of potential danger to a computer or network (that is, a vulnerability, which can be used in cyberattacks).

All of the program types stated below are capable of endangering user data or confidentiality. Programs that do not conceal their presence in the system (e.g. spam distribution software or traffic analyzers) are usually not considered computer threats, although they can also harm the user under certain circumstances.

Computer Viruses

Computer threats of this type are capable of embedding their code in other programs. Such embedding is called *infection*. In most cases, an infected file becomes a virus carrier and the embedded code does not necessarily match the original one. A significant number of viruses is designed to damage or destroy data.

In Doctor Web classification, viruses are separated by the type of objects they infect:

- *File viruses* infect files of the operating system (usually executable files and dynamic libraries) and are activated when an infected file is started.
- *Macro-viruses* infect documents used by Microsoft® Office (and other applications supporting macros, for example, written in Visual Basic). *Macros* are embedded programs written in a fully functional programming language and can be run in specific conditions (for instance, in Microsoft® Word, macros can be automatically started upon opening, closing or saving a document).
- *Script viruses* are created using script languages and usually infect other script files (e.g. service files of an operating system). They can also infect files of other types that allow execution of scripts and can spread, for example, via vulnerable web applications.
- *Boot viruses* infect boot records of disks and partitions as well as master boot records of hard drives. They do not require much memory and remain ready to continue performing their tasks until the system is unloaded, restarted or shut down.

Most viruses employ certain mechanisms of protection against detection. They are constantly being improved, and ways to cope with them are constantly being developed. All viruses can also be classified according to their type of protection against detection:

- *Encrypted viruses* encrypt their code upon every infection to hinder their detection in a file, a boot sector or memory. All instances of such viruses contain only a short common code fragment (the decryption procedure) that can be used as a signature.

- *Polymorphic viruses* not only encrypt their code, but they also generate a special decryption procedure that is different in every instance of the virus. As a result, such viruses do not have byte signatures.
- *Stealth viruses* (invisible viruses) perform certain actions to disguise their activity and to conceal their presence in infected objects. Such viruses gather the characteristics of an object before infecting it and then pass old data when the operating system or a program scans for modified files.

Viruses can also be classified according to a programming language in which they are written (for example, a low-level language such as an assembly language or a high-level language such as Go) or according to infected operating systems.

Computer Worms

Like viruses, programs of the *“computer worm”* type can copy themselves, but they do not infect other objects. A worm gets into a computer from a network (typically as an email attachment or from the internet) and sends its functioning copies to other computers. Worms either rely on user actions or spread automatically.

Worms do not necessarily consist of only one file (the worm body). Many of them have a so called infectious part (the shellcode), which loads into the computer operating memory and then downloads the worm body as an executable file over the network. Until the worm body is downloaded to the system, the worm can be avoided by rebooting the computer (at which the operating memory is reset). However, if the worm body infiltrates the system, then only an anti-virus program can cope with it.

Worms are capable of rendering entire networks inoperable even if these worms do not carry any payload (i.e. do not cause any direct damage to the system).

In Doctor Web classification, worms are separated by their distribution method (environment):

- *Network worms* spread via various network and file sharing protocols.
- *Mail worms* spread via email protocols (POP3, SMTP and so on).
- *Chat worms* spread via popular instant messaging services (ICQ, IM, IRC and so on).

Trojans

Malware of this type cannot reproduce itself. A trojan pretends to be a popular program and performs its functions (or imitates its operation). Meanwhile, it performs some malicious actions (damages or deletes data, sends confidential information and so on) or makes it possible for a hacker to access the computer without permission, for example, in order to harm a third party.

Trojan masking and malicious features are similar to those of a virus. A trojan can even be a component of a virus. However, most trojans spread as separate executable files (through file exchange servers, removable data carriers or email attachments) that are started by users or certain system processes.

It is very hard to classify trojans due to the fact that they are often spread by viruses and worms and also because many malicious actions that can be performed by other types of threats are attributed to trojans only. Here are some trojan types that are classified by Doctor Web as separate classes:

- *Backdoors* are trojans that allow to gain privileged access to a system, bypassing existing access and security measures. Backdoors do not infect files.
- *Rootkits* are used to intercept system functions of an operating system in order to conceal themselves. Furthermore, a rootkit can hide processes of other programs, directories and files. It can spread either as an independent program or as an auxiliary component of another malicious program. There are two kinds of rootkits according to their operation mode: *User Mode Rootkits—UMR* (intercept functions of user mode libraries) and *Kernel Mode Rootkits—KMR* (intercept functions at the system kernel level, which makes them harder to detect and neutralize).
- *Keyloggers* are used to log data that users enter by means of a keyboard. The aim of this is to steal personal information (i.e. network passwords, logins, primary account numbers and so on).
- *Clickers* redefine hyperlinks when they are clicked and thus redirect users to certain websites (sometimes malicious). This is usually done to increase ad traffic of websites or perform distributed denial-of-service (DDoS) attacks.
- *Proxy trojans* provide anonymous internet access to a malicious actor through a victim’s computer.

In addition, trojans can also change the start page in a web browser or delete certain files. However, these actions can also be performed by other types of threats (viruses and worms).

Hacktools

Hacktools are programs designed to assist an intruder with hacking. The most common among them are port scanners that detect vulnerabilities in firewalls and other components of a computer protection system. Besides hackers, such tools are used by administrators to test security of their networks. Occasionally, programs that use social engineering techniques are classified as hacktools.

Adware

Usually, this term refers to program code embedded in freeware programs that forces displaying of advertisements to a user. However, sometimes such code can be distributed via other malicious programs and display advertisements, for example, in web browsers. Many adware programs operate with data collected by spyware.

Jokes

Like adware, this type of malware cannot be used to inflict any direct damage to the system. Joke programs usually just generate messages about errors that never occurred and threaten to perform actions that will lead to data loss. Their purpose is to frighten or annoy users.

Dialers

These are special programs that are designed to scan a range of telephone numbers and find those where a modem answers. These numbers are then used to mark up the price of telephoning facilities or to connect the user to expensive telephone services.

Riskware

These programs were not created for malicious purposes, but due to their characteristics they can pose a threat to the system security. Riskware can accidentally damage or delete data or be used by malicious actors or other programs to harm the system. Riskware includes various remote chat and administrative tools, FTP servers and so on.

Suspicious Objects

Suspicious objects include any potential threats detected by a heuristic analyzer. Such objects can potentially relate to any type of computer threats (even unknown to information security specialists) or appear to be safe in case of false positives. It is recommended that files containing suspicious objects are quarantined and sent to Doctor Web anti-virus laboratory experts for analysis.

10.2. Appendix B. Neutralizing Computer Threats

In this appendix

- Detection Methods
- Threat-related Actions

All Doctor Web anti-virus solutions use a set of methods to detect threats, which allows to scan suspicious objects thoroughly.

Detection Methods

Signature Analysis

This method of detection is the first to run. It is applied by scanning object contents for known threat signatures. A signature is a continuous finite sequence of bytes necessary and sufficient to identify a threat unequivocally. At that, the search for signatures of the objects being scanned is performed using checksums, which allows to reduce virus databases significantly in size having preserved, at the same time, unequivocal matching and correct detection of threats and curing infected objects. Dr.Web virus databases are composed such that the same record can cover entire classes or families of threats.

Origins Tracing™

This unique Dr.Web technology allows to detect new and modified threats using already known infecting and damaging techniques covered by virus databases. The technology is used after completion of the signature analysis and protects users using Dr.Web anti-virus solutions against such threats as the notorious Trojan.Encoder.18 ransomware (also known as gpcode). Furthermore, using the Origins Tracing™ technology allows to considerably reduce the number of false positives of the heuristic analyzer. Objects detected using Origins Tracing™ have the `.Origin` postfix added to their names.

Execution Emulation

The technology of program code emulation is used for detection of polymorphic and encrypted viruses when a search by checksums cannot be performed directly, or is considerably complicated (for example, it is impossible to generate reliable signatures). The method consists in emulating the execution of an analyzed code with an *emulator*—a programming model of a processor and runtime environment. An emulator operates with a protected memory area (an *emulation buffer*). At that, instructions are not passed to a CPU for actual execution. If the code processed by the emulator is infected, the emulation results in restoring the original malicious code available for signature analysis.

Heuristic Analysis

The operation of the heuristic analyzer is based on a set of *heuristics*—assumptions about fingerprints of both malware and safe code, which statistical significance is proved experimentally. Each fingerprint has a certain *weight* (that is, a number which determines a level of its severity and reliability). The weight can be positive if the fingerprint is indicative of malicious behavior or negative if the fingerprint is non-typical for computer threats. Depending on the total weight of contents of an object, the heuristic analyzer calculates a probability of this object containing unknown malware. If a threshold is exceeded, the heuristic analyzer returns a verdict that the analyzed object is malicious.

The heuristic analyzer also uses the FLY-CODE™ technology, which is a versatile algorithm to unpack files. This technology allows to make heuristic assumptions about the presence of malware in objects packed not only by those packers that Dr.Web developers are aware of, but also by new, previously unknown programs. While scanning packed objects, their structural entropy is being analyzed, thereby allowing to detect threats on the basis of the allocation of their code segments. This technology allows to detect multiple different threats packed by the same polymorphous packer on the basis of a single record in a virus database.

As any system of hypothesis testing under uncertainty, the heuristic analyzer can make type I or type II errors (skipping unknown threats or raising false positives correspondingly). Thus, objects detected by the heuristic analyzer are treated as “suspicious”.

While performing any of the scans, Dr.Web anti-virus solutions use the most recent information about all known malware. Threat signatures, footprints and behavioral patterns are updated and added to virus databases as soon as experts of the Doctor Web anti-virus laboratory discover new threats, occasionally several times per hour. Even if the newest malicious program passes Dr.Web real-time protection and penetrates the system, this program will be detected in the list of processes and neutralized after updating virus databases.

Cloud-based Threat Detection Technologies

Cloud-based detection methods allow to check any object (a file, an application, a browser extension and so on) against its *hash sum*—a unique sequence of digits and letters of a given length. When checked against their hash sums, objects are searched in the existing database and then classified into categories: clean, suspicious, malicious and so on.

This technology reduces time required for file scanning and saves device resources. The verdict is almost instantaneous given that the hash sum and not the object itself is analyzed. If Dr.Web Cloud servers are unavailable, the files are scanned locally, and the cloud scanning is resumed when the connection is restored.

Thus, the Dr.Web Cloud service collects information from numerous users and quickly updates data on previously unknown threats thereby increasing the effectiveness of device protection.

Threat-related Actions

Dr.Web products implement a number of actions that can be applied to detected objects to neutralize computer threats. The user can keep default actions applied to specific types of threats automatically, adjust these actions or choose the required action manually each time upon detection. Available actions are provided below. The brackets contain a parameter denoting a corresponding action and used in the unified configuration file and commands of the Dr.Web Ctl tool.

- **Report** (`REPORT`)—report the threat without applying any action to the infected object.
- **Cure** (`CURE`)—attempt to cure the infected object by removing only malicious content from its body.

> This action can be applied only to certain types of threats.

- **Quarantine** (`QUARANTINE`)—put the infected object (if possible) in a specialized quarantine directory to isolate it.
- **Delete** (`DELETE`)—permanently delete the infected object.

> If a threat is detected in a file inside a container (an archive, an email message and so on), the container is quarantined and not deleted.

The following actions can be applied to email messages when Dr.Web MailD scans them:

- **Pass** (`PASS`)—skip the detected threat without applying any action.
- **Discard** (`DISCARD`)—accept the message, but do not deliver it to the recipient.
- **Reject** (`REJECT`)—reject the message and prevent its delivery to the recipient.
- **Tempfail** (`TEMPFAIL`)—do not deliver the email message, return an error message to the sender or the recipient instead.
- **Repack** (`REPACK`)—before delivery of the email message to the recipient, modify this message by isolating threats in an archive attached to it and add a threat detection notification to the email message.
- **Add header** (`ADD_HEADER`)—add a specified header to the email message and deliver it to the recipient.
- **Change header** (`CHANGE_HEADER`)—change the value of the specified header and deliver the message to the recipient.

10.3. Appendix C. Technical Support

If you have a problem installing or using Doctor Web products, please try the following before contacting technical support:

1. Download and review the latest manuals and guides at https://download.drweb.com/doc/.
2. See the Frequently Asked Questions section at https://support.drweb.com/show_faq/.
3. Browse the official Doctor Web forum at https://forum.drweb.com/.

If you haven't found a solution to your problem, you can request direct assistance from Doctor Web technical support specialists. Please use one of the options below:

1. Fill out a web form in the appropriate section at https://support.drweb.com/.
2. Call +7 (495) 789-45-86 (for customers in Moscow) or 8-800-333-79-32 (a toll-free line for customers within Russia).

For information on regional and international offices of Doctor Web, please visit the official website at https://company.drweb.com/contacts/offices/.

To facilitate processing of your issue, we recommend that you generate a data set for the installed product, its configuration, and system environment before contacting our technical support. To do that, you can use a special utility included in Dr.Web Security Space.

To collect data for our technical support, run the following command:

```
# <opt_dir>/bin/support-report.sh <path to file>
```

where:

- *<opt_dir>*—directory in which main files of Dr.Web Security Space are stored, including executable files and libraries (by default, `/opt/drweb.com` for GNU/Linux);
- *<path to file>*—path to an archive in the `.tgz` format to which the debugging information about the product and the system environment will be written, for example, `/tmp/report.tgz`. Already existing files will not be rewritten. If the path is not specified, the archive will be stored in the directory of the superuser who started the utility (for example, `/root`) and will be named as follows:

  ```
  drweb.report.<timestamp>.tgz
  ```

  where *<timestamp>* is a full timestamp of creating the report, down to milliseconds, for example: 20190618151718.23625.

For details on conventions used for directories, refer to the section Introduction.

The utility for collecting data for our technical support must be run with superuser (*root*) privileges. To get superuser privileges, run the `su` command to change the user or the `sudo` command to run software as another user.

During operation, the utility collects and archives the following information:

- information about your OS (name, architecture, output of the `uname -a` command);

- list of packages installed on your system, including Doctor Web packages;
- log contents:
  - Dr.Web Security Space logs (if configured for separate components);
  - log of the `syslog` system daemon (`/var/log/syslog`, `/var/log/messages`);
  - log of a system package manager (`apt`, `yum`, etc.);
  - `dmesg` log;
- output of the commands `df`, `ip a` (`ifconfig -a`), `ldconfig -p`, `iptables-save`, `nft export xml`.
- information about settings and configuration of Dr.Web Security Space:
  - list of downloaded virus databases (`drweb-ctl baseinfo -l`);
  - list of files from Dr.Web Security Space directories and MD5 hash values of these files;
  - Dr.Web Virus-Finding Engine version and MD5 hash value;
  - information about the user and permissions retrieved from the key file, if Dr.Web Security Space is not running in centralized protection mode.

10.4. Appendix D. Dr.Web Security Space Configuration File

Configuration parameters of all Dr.Web Security Space components are managed by the Dr.Web ConfigD configuration management daemon. These parameters are stored in the `drweb.ini` file whose default directory is `/etc/opt/drweb.com/`.

> The text configuration file stores only those parameters whose values differ from defaults. If a parameter is absent in the configuration file, its default value is used.

You can view the list of all available parameters, including those that are absent in the configuration file and have default values, by running the command:

```
$ drweb-ctl cfshow
```

You can change any parameter value in two ways:

- By running the following command:

```
# drweb-ctl cfset <section>.<parameter> <new value>
```

> This command requires to run the Dr.Web Ctl management tool with superuser (the *root* user) privileges. To obtain superuser privileges, use the `su` or `sudo` command.
>
> For further information about the syntax of the `cfshow` and `cfset` commands of the Dr.Web Ctl tool (the `drweb-ctl` module), refer to the Dr.Web Ctl section.

- By specifying this parameter in the configuration file (by editing the file in any text editor) and reloading the configuration of Dr.Web Security Space to apply changes with the command:

```
# drweb-ctl reload
```

10.4.1. File Structure

The configuration file complies with the rules indicated below.

- The file content is separated into named sections. Allowed section names are strictly set and cannot be arbitrary. A section name is specified in square brackets and is identical to the name of the Dr.Web Security Space component that uses parameters of this section (except for the `[Root]` section, which stores parameters of the Dr.Web ConfigD configuration daemon).
- The `;` or `#` characters in the beginning of the lines of the configuration file indicate a comment. Such lines are skipped by Dr.Web Security Space components while reading configuration parameters from the configuration file.

- Each line in the file can contain only one parameter:

```
<parameter> = <value>
```

- All parameter names are strictly set and cannot be arbitrary.
- All section and parameter names are case-insensitive. Parameter values, except for names of directories and files in paths (for UNIX-like OSes), are also case-insensitive.
- Parameter values in the configuration file must be enclosed in quotation marks if these values contain spaces.
- Some parameters can accept multiple values. If so, the values are either comma-separated or specified several times in different lines of the configuration file. In the former case, spaces around a comma are ignored. If a space character is a part of a parameter value, the entire value must be enclosed in quotation marks.

  You can specify multiple values as:

  - a comma-separated list:

    ```
    Parameter = "Value1", "Value2","Value 3"
    ```

  - a sequence of lines in the configuration file:

    ```
    Parameter = Value2
    Parameter = Value1
    Parameter = "Value 3"
    ```

  The order of values is unimportant.

  Paths to files are always enclosed in quotation marks when separated by commas, for example:

  ```
  ExcludedPaths = "/etc/file1", "/etc/file2"
  ```

For the description of the sections of the configuration file, see the description of the Dr.Web Security Space components using it.

10.4.2. Parameter Types

Configuration parameters can be of the following types:

1. *Address*—a network connection address specified as *<IP address>*:*<port>*. In some cases the port can be omitted (in each case this is indicated in the description of the parameter).
2. *Boolean*—a flag parameter accepting `On`, `Yes`, `True` (the parameter is enabled) and `Off`, `No`, `False` (the parameter is disabled).
3. *Integer*—a non-negative integer.
4. *Fractional number*—a non-negative number with a fractional part can be indicated as a parameter value.

5. *Time interval*—a time interval consisting of a non-negative integer and a suffix character indicating the specified unit of measurement. The following suffixes representing units of measurement can be used:

   - `w`—weeks (`1w` = `7d`);
   - `d`—days (`1d` = `24h`);
   - `h`—hours (`1h` = `60m`);
   - `m`—minutes (`1m` = `60s`);
   - `s` or no suffix—seconds.

   If the time interval is specified in seconds, you can specify milliseconds after a point (no more than three digits, for example, `0.5s`—500 milliseconds). It is possible to specify multiple time intervals in different time units as a single time interval; in this case, it is counted as a sum of intervals (in fact, the configuration parameters always store a number of milliseconds covered by this time interval).

   In general terms, any time interval can be represented by the following expression: $N_1\mathtt{w}N_2\mathtt{d}N_3\mathtt{h}N_4\mathtt{m}N_5[.N_6]\mathtt{s}$, where N_1, ..., N_6 is a number of the corresponding time unites included in this interval. For example, a year (365 days) can be represented as follows (all records are equivalent): `365d`, `52w1d`, `52w24h`, `51w7d24h`, `51w7d23h60m`, `8760h`, `525600m`, `31536000s`.

   The examples below show you how intervals of 30 minutes, 2 seconds, 500 milliseconds can be specified:

   - in the configuration file:

     ```
     UpdateInterval = 30m2.5s
     ```

   - using the `drweb-ctl cfset` command:

     ```
     # drweb-ctl cfset Update.UpdateInterval 1802.5s
     ```

   - via a command-line parameter (for example, for the scanning engine):

     ```
     $ drweb-se --WatchdogInterval 1802.5
     ```

6. *Size*—the parameter value represents a size of an object (a file, a buffer, a cache, and so on), specified as a non-negative integer and a suffix representing a measurement unit. The following suffixes that specify size units can be used:

   - `mb`—megabytes (`1mb` = `1024kb`);
   - `kb`—kilobytes (`1kb` = `1024b`);
   - `b`—bytes.

   If a suffix is omitted, the size is considered to be in bytes. A single size record can be specified by multiple sizes in different units; in this case, the resulting size is counted as their sum (in fact, the configuration parameters always store the size in bytes).

7. *Path to directory (file)*—the parameter value is a string representing an acceptable path to a directory (a file).

   The file path must be ended with a file name.

   On UNIX-like operating systems, names of directories and files are case-sensitive. If it is not explicitly mentioned in a parameter description, paths cannot be represented by masks with special characters (`?`, `*`).

8. *Logging level*—a level at which the Dr.Web Security Space component events are logged. The following values are allowed:

   - `DEBUG`—the most detailed logging level. All messages and debug information are logged.
   - `INFO`—all messages are logged.
   - `NOTICE`—all error messages, warnings, and notifications are logged.
   - `WARNING`—all error messages and warnings are logged.
   - `ERROR`—only error messages are logged.

9. *Log type*—the parameter value specifies the Dr.Web Security Space component logging method. The following values are allowed:

   - `Stderr[:ShowTimestamp]`—messages are output to *stderr* (standard error stream). This value can be used *only* in the settings of the configuration daemon. At that, if it works in the background (“*daemonized*”), i.e. it is started with the `-d` parameter, this value *cannot* be used because components operating in the background cannot access input/output streams of the terminal). The additional `ShowTimestamp` parameter instructs to add a timestamp to every message.
   - `Auto`—messages for logging are passed to the Dr.Web ConfigD configuration daemon, which stores them in a single location specified in its own settings (the `Log` parameter in the `[Root]` section). This value is specified for all components *except for the configuration daemon* and is used as a default value.
   - `Syslog[:<`*facility*`>]`—the component will pass messages to the `syslog` system logging service.
   - The <*facility*> additional option is used to specify a type of a log to store messages logged by `syslog`. The following values are allowed:
     - `DAEMON`—messages from daemons,
     - `USER`—messages from user processes,
     - `MAIL`—messages from mail programs,
     - `LOCAL0`—messages from local processes 0,

     …
     - `LOCAL7`—messages from local processes 7.

- <*path*>—messages will be stored by the component directly in the specified log.

Examples of how to specify a parameter value:

- in the configuration file:

```
Log = Stderr:ShowTimestamp
```

- using the `drweb-ctl cfset` command:

```
# drweb-ctl cfset Root.Log /var/opt/drweb.com/log/general.log
```

- via a command-line parameter (for example, for the scanning engine):

```
# /opt/drweb.com/bin/drweb-se <socket> --Log Syslog:DAEMON
```

10. *Action*—action to be applied by the Dr.Web Security Space component upon detection of certain threats or upon another event. Allowed values:

    - **Report** (`REPORT`)—only notify of threat detection without applying other actions;
    - **Cure** (`CURE`)—attempt to cure the threat (that is, remove only malicious content from the file body);
    - **Quarantine** (`QUARANTINE`)—put the infected file in quarantine;
    - **Delete** (`DELETE`)—delete the infected file.

    Some of the actions can be applied only upon certain events (for example, a “Scanning error” event cannot trigger the `CURE` action). Allowed actions are always listed in the description of each parameter of the *action* type.

Other types of parameters and their allowed values are explicitly specified in the description of these parameters.

10.5. Appendix E. Generating SSL certificates

For the Dr.Web Security Space components that use a secure SSL/TLS data channel and application protocols, such as HTTPS, LDAPS, SMTPS and so on, to exchange data, it is necessary to provide private SSL keys and the corresponding certificates. Keys and certificates for some components are generated automatically; as for the others, they should be provided by a Dr.Web Security Space user. All the components use certificates in the PEM format.

To generate private keys and certificates used for connections via SSL/TLS, including verification certificates of Certification Authority (CA) and signed certificates, you can use the `openssl` command-line utility (included in the OpenSSL cryptographic package).

Consider a sequence of actions required for generating a private key and the corresponding SSL certificate together with an SSL certificate signed with a CA verification certificate.

To generate a private SSL key and a certificate

1. To generate a private key (the RSA algorithm, the key length is 2048 bits), run the command:

   ```
   $ openssl genrsa -out keyfile.key 2048
   ```

   If you want to password protect the key, use the `–des3` option. The generated key is in the `keyfile.key` file located in the current directory.

   To view the generated key, use the command:

   ```
   $ openssl rsa -noout -text -in keyfile.key
   ```

2. To generate a certificate for a specified time period based on the existing private key (in this case, for 365 days), run the command:

   ```
   $ openssl req -new -x509 -days 365 -key keyfile.key -out certificate.crt
   ```

   This command will request data (a name, an organization and so on) that identify the certified object. The generated certificate will be located in the `certificate.crt` file.

   To scan the contents of the generated certificate, use the command:

   ```
   $ openssl x509 -noout -text -in certificate.crt
   ```

To register a certificate as a trusted CA certificate

1. Move or copy the certificate file to the system trusted certificate directory (`/etc/ssl/certs` on Debian or Ubuntu).
2. In the trusted certificate directory, create a symbolic link to the certificate, where the name of the link is the hash value of the certificate.
3. Reindex the contents of the system directory containing certificates.

The example commands provided below perform all these three actions. It is assumed that the current directory is the trusted certificate directory `/etc/ssl/certs` and the certificate that is registered as a trusted one is located in the `/home/user/ca.crt` file:

```
# cp /home/user/ca.crt .
# ln -s ca.crt `openssl x509 -hash -noout -in ca.crt`.0
# c_rehash /etc/ssl/certs
```

To create a signed certificate

1. Generate a request file for signing a certificate *(Certificate Signing Request—CSR)* based on an existing private key. If the key is absent, generate it.

   The request for signing is created with the command:

   ```
   $ openssl req -new -key keyfile.key -out request.csr
   ```

   This command, as well as the command for certificate creation, requests data that identifies the certified object. Here, `keyfile.key` is the existing file of the private key. The received request will be saved to the `request.csr` file.

   To check the result of request creation, use the command:

   ```
   $ openssl req -noout -text -in request.csr
   ```

2. To create a signed certificated based on the request and the existing CA certificate, use the command:

   ```
   $ openssl x509 -req -days 365 -CA ca.crt -CAkey ca.key -set_serial 01 -in
   request.csr -out sigcert.crt
   ```

   > To create a signed certificate, you must have the following three files: the file of the root certificate `ca.crt` and its private key `ca.key` (the `certificate.crt` certificate and the `keyfile.key` key may be used istead of `ca.crt` and `ca.key`, then the obtained certificate will be self-signed), as well as the request for signing `request.csr`. The created signed certificate will be saved to the file `sigcert.crt`.

   To check the result, use the command:

   ```
   $ openssl x509 -noout -text -in sigcert.crt
   ```

Repeat the procedure of creating a key and a certificate (or a signed certificate, if necessary) as many times as the number of unique certificates you need to create. For example, from a security point of view, every agent for distributed file scanning by Dr.Web Network Checker within a scanning cluster should have its own key/certificate pair.

Converting a signed certificate

Some browsers or mail clients may require to convert the signed certificate used for authorization to the `PKCS12` format.

Such conversion can be performed by using the command:

```
# openssl pkcs12 -export -in sigcert.crt -out sigcert.pfx -inkey keyfile.key
```

Here, `sigcert.crt` is the existing file of the signed certificate, `keyfile.key` is the file of the corresponding private key. The resulting converted certificate is saved to the `sigcert.pfx` file.

10.6. Appendix F. Building Kernel Module for SpIDer Guard

In this section:

- General Information.
- Building the Kernel Module.
- Possible Build Errors.

General Information

If the operating system does not support the fanotify mechanism used by SpIDer Guard to monitor operations on file system objects, it uses a custom loadable module (LKM) operating in kernel space.

By default, SpIDer Guard is distributed with a built kernel module for the operating systems that do not support the fanotify service. Moreover, you can build the loadable kernel module manually from the source code files distributed with SpIDer Guard in the `.tar.bz2` archive.

> The LKM used by SpIDer Guard is designed for Linux kernels of versions 2.6.* and later.
>
> The LKM is not supported for ARM64, E2K and IBM POWER (ppc64el) architectures.

The archive with source code files is stored in the `share/drweb-spider-kmod/src` subdirectory of the Dr.Web Security Space base directory (by default, `/opt/drweb.com`) and is named as follows: `drweb-spider-kmod-`*<version>*`-`*<date>*`.tar.bz2`. The `drweb-spider-kmod` directory also contains the `check-kmod-install.sh` test script. Run the script to check whether your operating system supports precompiled kernel module versions already included in Dr.Web Security Space. If not, a message prompting to manually build the module will be displayed on the screen.

If the specified directory `drweb-spider-kmod` does not exist, install the `drweb-spider-kmod` package.

> To build the LKM manually from source code, superuser privileges are required. For that purpose, you can use the `su` command to switch to another user or the `sudo` command to build the module as another user.

Building the Kernel Module

1. Unpack the archive with source code to any directory. For example, the command

```
# tar -xf drweb-spider-kmod-<version>-<date>.tar.bz2
```

unpacks the archive directly to the directory containing the archive itself, having created a subdirectory with the name of the archive file (note that superuser privileges are required to write to the directory with the archive).

2. Navigate to the created directory with source code and run the command:

   ```
   # make
   ```

   If errors occur at the step of *make*, resolve them (see below) and restart compilation.

3. After successfully passing the *make* step, run the commands:

   ```
   # make install
   # depmod
   ```

4. After the kernel module is successfully built and registered in the system, perform additional configuration of SpIDer Guard. Specify the mode in which the component operates with the kernel module by running the command:

   ```
   # drweb-ctl cfset LinuxSpider.Mode LKM
   ```

   It is also possible to specify `AUTO` instead of `LKM`. In this case, SpIDer Guard will attempt to use both the kernel module and the fanotify system mechanism. For details, run the command:

   ```
   $ man 1 drweb-spider
   ```

Possible Build Errors

Upon running the *make* command, errors may occur. In that event, check the following:

- To ensure successful building of the module, the Perl interpreter and the GCC compiler are required. If they are absent in the system, install them.
- On certain OSes, you may need to install the `kernel-devel` package in advance.
- On certain OSes, the procedure can fail because the path to the directory with kernel source code files was specified incorrectly. In that event, use the `make` command with the `KDIR=`<*path to kernel source code*> parameter. Typically, the source code files are stored in the `/usr/src/kernels/`<*kernel version*> directory.

The kernel version returned by the `uname -r` command can differ from the <*kernel version*> directory name.

10.7. Appendix G. Known Errors

In this section

- Recommendations for Error Identification
- Error Codes

- Errors Without Code

If the occurred error is not present in this section, it is recommended that you contact our technical support. Be ready to provide the error code and describe steps to reproduce the issue.

Recommendations for Error Identification

- To identify a possible cause and background of the error, refer to the Dr.Web Security Space log (by default, it is stored in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the command:

```
# drweb-ctl log
```

- To identify errors, we recommend that you store output of the log in a separate file and enable output of detailed debug information. For that, run the commands:

```
# drweb-ctl cfset Root.Log <log path>
# drweb-ctl cfset Root.DefaultLogLevel DEBUG
```

- To reset the logging settings to defaults, run the commands:

```
# drweb-ctl cfset Root.Log -r
# drweb-ctl cfset Root.DefaultLogLevel -r
```

Error Codes

Error message: *Error on monitor channel*

Error code: `x1`

Internally used name: `EC_MONITOR_IPC_ERROR`

Description: One or several components cannot connect to the Dr.Web ConfigD configuration daemon.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Restart the configuration daemon:

```
# service drweb-configd restart
```

2. Check whether the authentication mechanism for PAM is installed, configured and operates correctly. If necessary, install and configure it (for details refer to administration manuals for your OS distribution).
3. If PAM is configured correctly and restarting the configuration daemon does not help, reset Dr.Web Security Space settings to defaults.

To do this, clear the contents of the *<etc_dir>*/drweb.ini file (you are recommended to make a backup copy of the configuration file), for example, by running the commands:

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

Restart the configuration daemon after clearing the contents of the configuration file.

4. If it is not possible to start the configuration daemon, reinstall the drweb-configd package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Operation is already in progress*

Error code: x2

Internally used name: EC_ALREADY_IN_PROGRESS

Description: The operation is already in progress.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Wait for the operation to finish. If necessary, repeat the required action later.

If the error persists, contact our technical support and provide the error code.

Error message: *Operation is in pending state*

Error code: x3

Internally used name: EC_IN_PENDING_STATE

Description: An operation requested by the user is in a pending state (possibly, a network connection is currently being established or one of the components is starting and initializing, which may take a long time).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Wait for the operation to start. If necessary, repeat the required action later.

If the error persists, contact our technical support and provide the error code.

Error message: *Interrupted by user*

Error code: `x4`

Internally used name: `EC_INTERRUPTED_BY_USER`

Description: The action has been terminated by the user (possibly, it was taking a long time).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Repeat the required action later.

If the error persists, contact our technical support and provide the error code.

Error message: *Operation canceled*

Error code: `x5`

Internally used name: `EC_CANCELED`

Description: The action has been canceled.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Repeat the required action.

If the error persists, contact our technical support and provide the error code.

Error message: *IPC connection terminated*

Error code: `x6`

Internally used name: `EC_LINK_DISCONNECTED`

Description: An IPC connection to one of the Dr.Web Security Space components has been terminated (possibly, the component was shut down due to being idle or by a user request).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

If the operation did not finish, repeat it later. Otherwise, the connection termination is not an error.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid IPC message size*

Error code: `x7`

Internally used name: `EC_BAD_MESSAGE_SIZE`

Description: A message of an invalid size has been received during component inter-process communication (IPC).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Restart Dr.Web Security Space:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid IPC message format*

Error code: `x8`

Internally used name: `EC_BAD_MESSAGE_FORMAT`

Description: A message of an invalid format has been received during component inter-process communication (IPC).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Restart Dr.Web Security Space:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Not ready*

Error code: `x9`

Internally used name: `EC_NOT_READY`

Description: The required action cannot be performed because the necessary component or device has not been initialized yet.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Repeat the required action later.

If the error persists, contact our technical support and provide the error code.

Error message: *Component is not installed*

Error code: `x10`

Internally used name: `EC_NOT_INSTALLED`

Description: The component necessary for running the required function is not installed.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Install or reinstall the necessary component. If you do not know the component name, try to determine it from the log file.
2. If the installation or reinstallation of the necessary component does not help, reinstall Dr.Web Security Space.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Unexpected IPC message*

Error code: `x11`

Internally used name: `EC_UNEXPECTED_MESSAGE`

Description: An invalid message has been received during component inter-process communication (IPC).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Restart Dr.Web Security Space by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *IPC protocol violation*

Error code: x12

Internally used name: EC_PROTOCOL_VIOLATION

Description: A protocol violation has occurred during component inter-process communication (IPC).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Restart Dr.Web Security Space by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Subsystem state is unknown*

Error code: x13

Internally used name: EC_UNKNOWN_STATE

Description: The required operation cannot be performed because a subsystem of Dr.Web Security Space is in an unknown state.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Repeat the operation.
2. If the error persists, restart Dr.Web Security Space by running the command:

   ```
   # service drweb-configd restart
   ```

   and repeat the operation afterwards.

If the error persists, contact our technical support and provide the error code.

Error message: *Path must be absolute*

Error code: x20

Internally used name: EC_NOT_ABSOLUTE_PATH

Description: A relative path to a file or a directory has been specified, whereas an absolute path is required.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Make the file or directory path absolute and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Not enough memory*

Error code: `x21`

Internally used name: `EC_NO_MEMORY`

Description: Not enough memory to complete the requested operation.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Increase the size of the memory available for Dr.Web Security Space processes (for example, by changing the limits using the `ulimit` command), restart Dr.Web Security Space and repeat the operation.

In some cases the `systemd` system service can ignore the specified limit changes. In this case, edit (or create if it does not exist) the file `/etc/systemd/system/drweb-configd.service.d/limits.conf` and indicate the changed limit value in it, for example:

```
[Service]
LimitDATA=32767
```

Available `systemd` limits can be determined using the `man systemd.exec` command.

Restart Dr.Web Security Space by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *I/O error*

Error code: `x22`

Internally used name: `EC_IO_ERROR`

Description: An input/output error has occurred (for example, a disk device has not been initialized yet or a partition of the file system is not available anymore).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the

OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Ensure the availability of the required I/O device or partition of the file system and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *No such file or directory*

Error code: `x23`

Internally used name: `EC_NO_SUCH_ENTRY`

Description: An attempt to access a non-existent file or directory has been made.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the indicated path. If necessary, correct it and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Permission denied*

Error code: `x24`

Internally used name: `EC_PERMISSION_DENIED`

Description: Insufficient privileges to access the specified file or directory.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Ensure that the path is correct and the component has the required privileges. If access to the object is denied, change its permissions or elevate component privileges and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Not a directory*

Error code: `x25`

Internally used name: `EC_NOT_A_DIRECTORY`

Description: The specified object of the file system is not a directory.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the object path. Specify the correct path and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Data file corrupted*

Error code: `x26`

Internally used name: `EC_DATA_CORRUPTED`

Description: The requested data is corrupted.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Repeat the operation.
2. If the error persists, restart Dr.Web Security Space:

   ```
   # service drweb-configd restart
   ```

   and repeat the operation afterwards.

If the error persists, contact our technical support and provide the error code.

Error message: *File already exists*

Error code: `x27`

Internally used name: `EC_FILE_EXISTS`

Description: A file with the same name already exists.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the file path. If necessary, correct it and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Read-only file system*

Error code: x28

Internally used name: EC_READ_ONLY_FS

Description: The file system is read-only.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Check the object path. Change it so as to indicate a writable partition of the file system and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Network error*

Error code: x29

Internally used name: EC_NETWORK_ERROR

Description: A network error has occurred (possibly, a remote host stopped responding unexpectedly or required connection failed).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Check that the network is available and network settings are correct. If necessary, change the network settings and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Not a drive*

Error code: x30

Internally used name: EC_NOT_A_DRIVE

Description: The input/output device being accessed is not a disk device.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Check the specified device name. Correct the path so as to indicate a disk device and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Unexpected EOF*

Error code: `x31`

Internally used name: `EC_UNEXPECTED_EOF`

Description: The end of the file has been reached enexpectedly while reading data.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the specified file name. If necessary, correct the path so as to indicate the correct file and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *File was changed*

Error code: `x32`

Internally used name: `EC_FILE_WAS_CHANGED`

Description: The file being scanned has been modified.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Repeat scanning.

If the error persists, contact our technical support and provide the error code.

Error message: *Not a regular file*

Error code: `x33`

Internally used name: `EC_NOT_A_REGULAR_FILE`

Description: The file system object being accessed is not a regular file (it may a directory, a socket and so on).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the specified file name. If necessary, change the path so as to indicate a regular file and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Name already in use*

Error code: `x34`

Internally used name: `EC_NAME_ALREADY_IN_USE`

Description: A file with the same name already exists.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the specified path. Correct it and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Host is offline*

Error code: `x35`

Internally used name: `EC_HOST_OFFLINE`

Description: A remote host cannot be accessed over the network.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check that the required host is available. If necessary, correct the host address and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Resource limit reached*

Error code: `x36`

Internally used name: `EC_LIMIT_REACHED`

Description: A limit on resource usage has been reached.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Check the availability of the required resource. If necessary, raise the limit on the usage of this resource and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Mounting points are different*

Error code: `x37`

Internally used name: `EC_CROSS_DEVICE_LINK`

Description: Restoring the file implies moving it between two different mount points.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Choose another path for restoring the file and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Unpacking error*

Error code: `x38`

Internally used name: `EC_UNPACKING_ERROR`

Description: Unable to unpack an archive (it may be password-protected or corrupted).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Make sure that the archive is not corrupted. If the archive is protected with a password, remove the protection having entered the correct password and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Virus database corrupted*

Error code: `x40`

Internally used name: `EC_BASE_CORRUPTED`

Description: Virus databases are corrupted.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the virus database directory. Change the path, if necessary (the `VirusBaseDir` parameter in the `[Root]` section of the configuration file). You can use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Root.VirusBaseDir
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Root.VirusBaseDir <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Root.VirusBaseDir -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *Non-supported virus database version*

Error code: `x41`

Internally used name: `EC_OLD_BASE_VERSION`

Description: Current virus databases are intended for an earlier program version.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the virus database directory. Change the path, if necessary (the `VirusBaseDir` parameter in the `[Root]` section of the configuration file). You can use the commands of the command-line management tool.

To get the current parameter value, run the command:

```
$ drweb-ctl cfshow Root.VirusBaseDir
```

To set a new parameter value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. Update virus databases: run the command:

```
$ drweb-ctl update
```

If the error persists, contact our technical support and provide the error code.

Error message: *Empty virus database*

Error code: `x42`

Internally used name: `EC_EMPTY_BASE`

Description: The virus database is empty.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the virus database directory. Change the path, if necessary (the `VirusBaseDir` parameter in the `[Root]` section of the configuration file). You can use the commands of the command-line management tool.

   To get the current parameter value, run the command:

```
$ drweb-ctl cfshow Root.VirusBaseDir
```

   To set a new parameter value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

   To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. Update virus databases: run the command:

```
$ drweb-ctl update
```

If the error persists, contact our technical support and provide the error code.

Error message: *Object cannot be cured*

Error code: x43

Internally used name: EC_CAN_NOT_BE_CURED

Description: The **Cure** action has been applied to an incurable object.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Select an action that can be applied to this object and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Non-supported virus database combination*

Error code: x44

Internally used name: EC_INVALID_BASE_SET

Description: Incompatible virus databases.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Check the path to the virus database directory. Change the path, if necessary (the VirusBaseDir parameter in the [Root] section of the configuration file). You can use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Root.VirusBaseDir
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Root.VirusBaseDir <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Root.VirusBaseDir -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *Scan limit reached*

Error code: `x45`

Internally used name: `EC_SCAN_LIMIT_REACHED`

Description: The specified limits have been exceeded during the scanning of an object (for example, on the size of an unpacked file, on the nesting depth and so on).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Change the scanning limits (in the component settings) using the `drweb-ctl cfshow` and `drweb-ctl cfset` commands.
2. After changing the settings, repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Authentication failed*

Error code: `x47`

Internally used name: `EC_AUTH_FAILED`

Description: Invalid user credentials have been used for authentication.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Provide valid credentials of a user having required privileges and try to authenticate again.

If the error persists, contact our technical support and provide the error code.

Error message: *Authorization failed*

Error code: `x48`

Internally used name: `EC_NOT_AUTHORIZED`

Description: The current user has insufficient privileges for performing the requested operation.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Provide valid credentials of a user having required privileges and try to authorize again.

If the error persists, contact our technical support and provide the error code.

Error message: *Access token is invalid*

Error code: `x49`

Internally used name: `EC_INVALID_TOKEN`

Description: A component of Dr.Web Security Space has provided an invalid authorization token in an attempt to execute an operation requiring elevated privileges.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Authenticate by providing valid credentials of a user having required privileges and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid argument*

Error code: `x60`

Internally used name: `EC_INVALID_ARGUMENT`

Description: Unable to run the command since an invalid argument has been provided.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the command syntax and format.
2. Repeat the required action having indicated a valid argument.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid operation*

Error code: `x61`

Internally used name: `EC_INVALID_OPERATION`

Description: An attempt to run an invalid command has been made.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Repeat the required action using a valid command.

If the error persists, contact our technical support and provide the error code.

Error message: *Superuser privileges required*

Error code: x62

Internally used name: EC_ROOT_ONLY

Description: Superuser privileges are required to perform the required action.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Elevate you privileges to superuser ones and repeat the required action. To elevate the privileges, use the su or sudo command.

If the error persists, contact our technical support and provide the error code.

Error message: *Not allowed in centralized protection mode*

Error code: x63

Internally used name: EC_STANDALONE_MODE_ONLY

Description: The required action can be performed only when Dr.Web Security Space operates in standalone mode.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Switch Dr.Web Security Space to the standalone mode and repeat the operation.

To switch Dr.Web Security Space to the standalone mode run the command:

```
# drweb-ctl esdisconnect
```

If the error persists, contact our technical support and provide the error code.

Error message: *Non-supported OS*

Error code: x64

Internally used name: EC_NON_SUPPORTED_OS

Description: Dr.Web Security Space does not support the operating system installed on the host.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Install an operating system from the list provided in system requirements.

If the error persists, contact our technical support and provide the error code.

Error message: *Feature not implemented*

Error code: `x65`

Internally used name: `EC_UNKNOWN_OPTION`

Description: The required features of the component are not implemented in the current version.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Try to reset the settings of Dr.Web Security Space to defaults.

To do this, clear the contents of the <*etc_dir*>`/drweb.ini` file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

Restart Dr.Web Security Space after clearing the contents of the configuration file by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Unknown option*

Error code: `x66`

Internally used name: `EC_UNKNOWN_OPTION`

Description: The configuration file contains parameters that are unknown to or not supported by the current version of Dr.Web Security Space.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Open the <*etc_dir*>/drweb.ini file in any text editor, remove the line containing the invalid parameter. Save the file and restart the Dr.Web ConfigD configuration daemon by running the command:

   ```
   # service drweb-configd restart
   ```

2. If this does not help, reset Dr.Web Security Space settings to defaults.

   To do this, clear the contents of the <*etc_dir*>/drweb.ini file (it is recommended to make a backup copy of the configuration file first), for example, by running the commands:

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   Restart the configuration daemon after clearing the contents of the configuration file.

If the error persists, contact our technical support and provide the error code.

Error message: *Unknown section*

Error code: x67

Internally used name: EC_UNKNOWN_SECTION

Description: The configuration file contains sections unknown to or not supported by the current version of Dr.Web Security Space.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Open the <*etc_dir*>/drweb.ini file in any text editor, remove the unknown (unsupported) section. Save the file and restart the Dr.Web ConfigD configuration daemon by running the command:

   ```
   # service drweb-configd restart
   ```

2. If this does not help, reset Dr.Web Security Space settings to defaults.

   To do this, clear the contents of the <*etc_dir*>/drweb.ini file (it is recommended to make a backup copy of the configuration file first), for example, by running the commands:

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   Restart the configuration daemon after clearing the contents of the configuration file.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid option value*

Error code: `x68`

Internally used name: `EC_INVALID_OPTION_VALUE`

Description: One or more parameters in the configuration file have invalid values.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Set the parameter value using the `drweb-ctl cfshow` and `drweb-ctl cfset` commands.

   If you do not know which value is valid for the parameter, refer to a man page for the component which uses this parameter. You can also restore a default value of the parameter.

2. You can also directly edit the <*etc_dir*>`/drweb.ini` configuration file. To do this, open it in any text editor, find the line containing an invalid parameter value, set a valid value, then save the file and restart the Dr.Web ConfigD configuration daemon by running the command:

   ```
   # service drweb-configd restart
   ```

3. If the previous steps did not help, reset Dr.Web Security Space settings to defaults.

   To do this, clear the contents of the <*etc_dir*>`/drweb.ini` file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   Restart the configuration daemon after clearing the contents of the configuration file.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid state*

Error code: `x69`

Internally used name: `EC_INVALID_STATE`

Description: Dr.Web Security Space or one of its components is in an invalid state and cannot complete the required operation.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Repeat the required action later.
2. If the error persists, restart Dr.Web Security Space by running the command:

   ```
   # service drweb-configd restart
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *Only one value allowed*

Error code: `x70`

Internally used name: `EC_NOT_LIST_OPTION`

Description: A list of values is set for a single-valued parameter in the configuration file.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Set the parameter value using the `drweb-ctl cfshow` and `drweb-ctl cfset` commands.

   If you do not know which value is valid for the parameter, refer to a man page for the component which uses this parameter. You can also restore a default value of the parameter.

2. You can also directly edit the <*etc_dir*>`/drweb.ini` configuration file. To do this, open it in any text editor, find the line containing an invalid parameter value, set a valid value, then save the file and restart the Dr.Web ConfigD configuration daemon by running the command:

   ```
   # service drweb-configd restart
   ```

3. If the previous steps did not help, reset Dr.Web Security Space settings to defaults.

   To do this, clear the contents of the <*etc_dir*>`/drweb.ini` file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   Restart the configuration daemon after clearing the contents of the configuration file.

If the error persists, contact our technical support and provide the error code.

Error message: *Record not found*

Error code: `x80`

Internally used name: `EC_RECORD_NOT_FOUND`

Description: The threat record is missing (it may have already been proccessed by another component).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Update the threat list later.

If the error persists, contact our technical support and provide the error code.

Error message: *Record is in process now*

Error code: x81

Internally used name: EC_RECORD_BUSY

Description: The threat is already being processed by another component.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Update the threat list later.

If the error persists, contact our technical support and provide the error code.

Error message: *File has already been quarantined*

Error code: x82

Internally used name: EC_QUARANTINED_FILE

Description: The file is already in quarantine. Probably, another component has already processed the threat.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Update the threat list later.

If the error persists, contact our technical support and provide the error code.

Error message: *Update zone is not provided by cloud*

Error code: x83

Internally used name: EC_NO_ZONE_IN_CLOUD

Description: An attempt to update using Dr.Web Cloud was unsuccessful.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Repeat the required action later.

If the error persists, contact our technical support and provide the error code.

Error message: *Update zone is not provided on disk*

Error code: `x84`

Internally used name: `EC_NO_ZONE_ON_DISK`

Description: An attempt to update the virus bases in offline mode was unsuccessful.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Make sure that the path to the device used for updating is correct.
2. Make sure that the user as whom you are trying to update has read permissions for the directory with updates.

If the error persists, contact our technical support and provide the error code.

Error message: *Cannot backup before update*

Error code: `x89`

Internally used name: `EC_BACKUP_FAILED`

Description: Prior to downloading updates from an update server, an attempt to make a backup copy of the files to be updated has failed.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the directory that stores backup copies of the files to be updated. Change the path, if necessary (the `BackupDir` parameter in the `[Update]` section of the configuration file).
   - To view and change the path, use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Update.BackupDir
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Update.BackupDir <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Update.BackupDir -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

3. If the error persists, check whether the user as whom the component is running has write

permissions for a directory specified in the BackupDir parameter. The name of this user is specified in the RunAsUser parameter. If necessary, change the user specified in the RunAsUser parameter or grant the lacking permissions in the directory attributes.

4. If the error persists, reinstall the drweb-update package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid DRL file*

Error code: x90

Internally used name: EC_BAD_DRL_FILE

Description: The integrity of one of the files storing a list of update servers is violated.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Check the path to the file storing the list of servers and change the path if necessary (parameters containing *DrlDir in the [Update] section of the configuration file). Use the commands of the command-line management tool.

   To view the current parameter value, run the command (replace <*DrlDir> with a specific parameter name. If the parameter name is unknown, view all parameter values in the section, skipping the command part in square brackets):

   ```
   $ drweb-ctl cfshow Update[.<*DrlDir>]
   ```

   To set a new parameter value, run the command (replace <*DrlDir> with a specific parameter name):

   ```
   # drweb-ctl cfset Update.<*DrlDir> <new path>
   ```

   To restore the default parameter value, run the command (replace <*DrlDir> with a specific parameter name):

   ```
   # drweb-ctl cfset Update.<*DrlDir> -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

3. If the error persists, reinstall the drweb-update package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid LST file*

Error code: x91

Internally used name: EC_BAD_LST_FILE

Description: The integrity of the file storing a list of virus databases to be updated is violated.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Update virus databases again after some time:run the command:

```
$ drweb-ctl update
```

2. If the error persists, reinstall the drweb-update package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid compressed file*

Error code: x92

Internally used name: EC_BAD_LZMA_FILE

Description: The integrity of the downloaded file containing updates is violated.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Update virus databases again after some time:run the command:

```
$ drweb-ctl update
```

If the error persists, contact our technical support and provide the error code.

Error message: *Proxy authentication error*

Error code: x93

Internally used name: EC_PROXY_AUTH_ERROR

Description: Unable to connect to update servers via the proxy server specified in the settings.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the

OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the parameters used to connect to the proxy server (set with the `Proxy` parameter in the `[Update]` section of the configuration file).

   - Use commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Update.Proxy
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Update.Proxy <new parameters>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Update.Proxy -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *No update servers available*

Error code: `x94`

Internally used name: `EC_NO_UPDATE_SERVERS`

Description: Unable to connect to any of the update servers.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check if the network is available. Change network settings if necessary.
2. If you can access the network only using a proxy server, configure connection to the proxy server (the `Proxy` parameter in the `[Update]` section of the configuration file).

   - Use commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Update.Proxy
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Update.Proxy <new parameters>
   ```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Update.Proxy -r
```

3. If the network connection parameters (including the parameters of the proxy server) are correct, but the error persists, make sure that you use a relevant list of update servers. The list of used update servers is set in `*DrlDir` parameters in the `[Update]` section of the configuration file.

   > If `*CustomDrlDir` parameters indicate an existing valid file storing a list of servers, the servers specified there will be used instead of the servers of the standard update zone (the value specified in the corresponding `*DrlDir` parameter is ignored).

   - Use the commands of the command-line management tool.

   To view the current parameter value, run the command (replace <**DrlDir*> with a specific parameter name. If the parameter name is unknown, view all parameter values in the section, skipping the command part in square brackets):

   ```
   $ drweb-ctl cfshow Update[.<*DrlDir>]
   ```

   To set a new parameter value, run the command (replace <**DrlDir*> with a specific parameter name):

   ```
   # drweb-ctl cfset Update.<*DrlDir> <new path>
   ```

   To restore the default parameter value, run the command (replace <**DrlDir*> with a specific parameter name):

   ```
   # drweb-ctl cfset Update.<*DrlDir> -r
   ```

4. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid key file format*

Error code: `x95`

Internally used name: `EC_BAD_KEY_FORMAT`

Description: The key file format is violated.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check that you have a key file and a correct path to it. You can specify the path to the key file in the `KeyPath` parameter in the `[Root]` section of the configuration file. Use the commands of the command-line management tool.

To get the current parameter value, run the command:

```
$ drweb-ctl cfshow Root.KeyPath
```

To set a new parameter value, run the command:

```
# drweb-ctl cfset Root.KeyPath <path to file>
```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Root.KeyPath -r
```

2. If you do not have a key file or the key file being used is corrupted, purchase and install it. For more details on the key file, purchasing and installing it, refer to the Licensing section.
3. You can view current license options on the **My Dr.Web** user webpage at https://support.drweb.com/get+cabinet+link/.

If the error persists, contact our technical support and provide the error code.

Error message: *License is already expired*

Error code: x96

Internally used name: EC_EXPIRED_KEY

Description: The license has expired.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Purchase a new license and install a received key file. For more details on how to purchase the license and install the key file, refer to the Licensing section.
2. You can view current license options on the **My Dr.Web** user webpage at https://support.drweb.com/get+cabinet+link/.

If the error persists, contact our technical support and provide the error code.

Error message: *Network operation timed out*

Error code: x97

Internally used name: EC_NETWORK_TIMEDOUT

Description: A network connection timed out (possibly, a remote host has stopped responding unexpectedly or the required connection has failed).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Check that the network is available and network settings are correct. If necessary, change the network settings and repeat the operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid checksum*

Error code: `x98`

Internally used name: `EC_BAD_CHECKSUM`

Description: The checksum of the downloaded file with updates is invalid.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

Update virus databases again after some time:run the command:

```
$ drweb-ctl update
```

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid trial license*

Error code: `x99`

Internally used name: `EC_BAD_TRIAL_KEY`

Description: The demo key file is invalid (for example, it was received for another computer).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Request a new demo period for this computer or purchase a new license and install a received key file. For more details on how to purchase the license and install the key file, refer to the Licensing section.
2. You can view current license options on the **My Dr.Web** user webpage at https://support.drweb.com/get+cabinet+link/.

If the error persists, contact our technical support and provide the error code.

Error message: *Blocked license key*

Error code: `x100`

Internally used name: `EC_BLOCKED_LICENSE`

Description: The current license is blocked (the terms of use of Dr.Web Security Space may be violated).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Purchase a new license and install a received key file. For more details on how to purchase the license and install the key file, refer to the Licensing section.
2. You can view current license options on the **My Dr.Web** user webpage at https://support.drweb.com/get+cabinet+link/.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid license*

Error code: `x101`

Internally used name: `EC_BAD_LICENSE`

Description: The license is intended for another product or does not cover Dr.Web Security Space components.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Purchase a new license and install a received key file. For more details on how to purchase the license and install the key file, refer to the Licensing section.
2. You can view current license options on the **My Dr.Web** user webpage at https://support.drweb.com/get+cabinet+link/.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid configuration*

Error code: `x102`

Internally used name: `EC_BAD_CONFIG`

Description: A component of Dr.Web Security Space cannot operate because of invalid configuration settings.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. If you do not know the name of the component that causes the error, try to determine it by viewing the log file.
2. If the error is caused by the SpIDer Guard component, the mode that is selected for the component operation is probably not supported by your operating system. Check the selected component mode and change it, if necessary, by setting the *Auto* value (the `Mode` parameter in the `[LinuxSpider]` section of the configuration file).

   - Use commands of the command-line management tool.

   To set the value to *Auto*, run the command:

   ```
   # drweb-ctl cfset LinuxSpider.Mode Auto
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset LinuxSpider.Mode -r
   ```

   If the error persists, build and install the loadable kernel module for SpIDer Guard manually.

   > ⚠ Operation of SpIDer Guard and of the loadable kernel module is guaranteed only if your OS is on the list of supported systems (see the System Requirements and Compatibility section).

3. If this error is caused by Dr.Web Firewall for Linux, most likely there is a conflict with another firewall. For example, it is known that Dr.Web Firewall for Linux conflicts with FirewallD on Fedora, CentOS, Red Hat Enterprise Linux (every time it starts, FirewallD corrupts traffic routing rules set by Dr.Web Firewall for Linux).

   To resolve this error, restart Dr.Web Security Space by running the command:

   ```
   # service drweb-configd restart
   ```

   or

   ```
   # drweb-ctl reload
   ```

   > If you allow FirewallD to operate, this error caused by Dr.Web Firewall for Linux can repeatedly occur on every restart of FirewallD, including the restart of the OS. You can resolve this error by disabling FirewallD (refer to the manual of FirewallD included in the manual of your OS).

4. If the error is caused by another component, reset its settings to defaults using any of the following methods:

   - using the `drweb-ctl cfshow` and `drweb-ctl cfset` commands;
   - edit the configuration file manually (delete all parameters from the component section).

5. If the previous steps did not help, reset Dr.Web Security Space settings to defaults.

To do this, clear the contents of the <*etc_dir*>`/drweb.ini` file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

Restart Dr.Web Security Space after clearing the configuration file by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid executable file*

Error code: `x104`

Internally used name: `EC_BAD_EXECUTABLE`

Description: Failed to start the component. The executable file is corrupted or the file path is invalid.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. If you do not know the name of the component that causes the error, try to determine it by viewing the log file.
2. Check the executable path to the component in the Dr.Web Security Space configuration file (the `ExePath` parameter in the component section) by running the command (replace <*component section*> with the name of the corresponding section of the configuration file):

   ```
   $ drweb-ctl cfshow <component section>.ExePath
   ```

3. Reset the path to its default value by running the command (replace <*component section*> with the name of the corresponding section of the configuration file):

   ```
   # drweb-ctl cfset <component section>.ExePath -r
   ```

4. If the previous steps did not help, reinstall the package of the corresponding component.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Core engine is not available*

Error code: `x105`

Internally used name: `EC_NO_CORE_ENGINE`

Description: The file of Dr.Web Virus-Finding Engine is missing or unavailable.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the anti-virus engine file `drweb32.dll` and, if necessary, correct it (the `CoreEnginePath` parameter in the `[Root]` section of the configuration file).

   - Use commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Root.CoreEnginePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Root.CoreEnginePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Root.CoreEnginePath -r
   ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

3. If the path is correct and the error persists after updating virus databases, reinstall the `drweb-bases` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *No virus databases*

Error code: `x106`

Internally used name: `EC_NO_VIRUS_BASES`

Description: Virus databases are missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the virus database directory. Change the path, if necessary (the `VirusBaseDir` parameter in the `[Root]` section of the configuration file). You can use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Root.VirusBaseDir
   ```

To set a new parameter value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. Update virus databases: run the command:

```
$ drweb-ctl update
```

If the error persists, contact our technical support and provide the error code.

Error message: *Process terminated by signal*

Error code: x107

Internally used name: EC_APP_TERMINATED

Description: A component has been shut down (possibly, owing to being idle or by a user request).

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. If the operation has not finished, repeat it. Otherwise, the shutdown is not an error.
2. If the component shuts down constantly, reset its settings to default using any of the following methods:
    - using the drweb-ctl cfshow and drweb-ctl cfset commands;
    - edit the configuration file manually (delete all parameters from the component section).
3. If this did not help, reset Dr.Web Security Space settings to default.

   To do this, clear the contents of the <*etc_dir*>/drweb.ini file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

   Restart Dr.Web Security Space after clearing the contents of the configuration file by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Error message: *Unexpected process termination*

Error code: `x108`

Internally used name: `EC_APP_CRASHED`

Description: A component has been shut down because of a failure.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Repeat the terminated operation.
2. If the component constantly shuts down abnormally, reset its settings to default using any of the following methods:
   - using the `drweb-ctl cfshow` and `drweb-ctl cfset` commands;
   - edit the configuration file manually (delete all parameters from the component section).
3. If this did not help, reset Dr.Web Security Space settings to default.

   To do this, clear the contents of the <*etc_dir*>`/drweb.ini` file (it is recommended that you make a backup of the configuration file), for example, by running the commands:

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   Restart Dr.Web Security Space after clearing the contents of the configuration file by running the command:

   ```
   # service drweb-configd restart
   ```

4. If the error persists after resetting Dr.Web Security Space settings, reinstall the component package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Incompatible software detected*

Error code: `x109`

Internally used name: `EC_INCOMPATIBLE`

Description: One or several components of Dr.Web Security Space cannot operate properly. Their operation is impeded by software in your system.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. If this error is caused by SpIDer Gate, the issue may be related to software generating rules for the NetFilter system firewall, which prevent SpIDer Gate from operating properly, such as Shorewall or SuseFirewall2 (on SUSE Linux). Such applications recurrently check the integrity of rule sets specified by them and rewrite these rules.

Reconfigure conflicting software so that it does not interfere in SpIDer Gate operation. If unsuccessful, disable the software so that it does not load at the operating system startup. You can try to configure SuseFirewall2 (on SUSE Linux) as follows:

1) open the configuration file of SuseFirewall2 (by default, it is `/etc/sysconfig/SuSEfirewall2`);

2) find the following lines in the file:

```
## Type: yesno
#
# Install NOTRACK target for interface lo in the raw table. Doing so
# speeds up packet processing on the loopback interface. This breaks
# certain firewall setups that need to e.g. redirect outgoing
# packets via custom rules on the local machine.
#
# Defaults to "yes" if not set
#
FW_LO_NOTRACK=""
```

3) Set the `FW_LO_NOTRACK` parameter value to `no`:

```
FW_LO_NOTRACK="no"
```

4) Restart SuseFirewall2:

```
# rcSuSEfirewall2 restart
```

⚠ If there is no `FW_LO_NOTRACK` parameter in SuseFirewall2 settings, stop this application and disable its launch at system startup to deter conflict.

After reconfiguring or disabling the conflicting application, restart SpIDer Gate.

2. If the error is caused by another component, disable or reconfigure the conflicting software such as to prevent any interference in the Dr.Web Security Space operation.

If the error persists, contact our technical support and provide the error code.

Error message: *Invalid anti-spam library*

Error code: `x110`

Internally used name: `EC_BAD_ANTISPAM_LIB`

Description: A file of the anti-spam library required for e-mail scanning is missing, unavailable or corrupted.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the library file and, if necessary, correct it (the `AntispamCorePath` parameter in the `[Root]` section of the configuration file).

- Use commands of the command-line management tool.

  To get the current parameter value, run the command:

  ```
  $ drweb-ctl cfshow Root.AntispamCorePath
  ```

  To set a new parameter value, run the command:

  ```
  # drweb-ctl cfset Root.AntispamCorePath <new path>
  ```

  To reset the parameter to its default value, run the command:

  ```
  # drweb-ctl cfset Root.AntispamCorePath -r
  ```

2. Update virus databases: run the command:

   ```
   $ drweb-ctl update
   ```

3. If the path is correct and the error persists after updating virus databases, reinstall the `drweb-maild` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *No web resource databases*

Error code: x112

Internally used name: EC_NO_DWS_BASES

Description: Databases of web resource categories are missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Check the path directory to a database of web resource categories. Change the path if necessary (the DwsDir parameter in the [Root] section of the configuration file).

   - Use commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow Root.DwsDir
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset Root.DwsDir <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset Root.DwsDir -r
   ```

2. Update databases of web resource categories:

   - run the command:

   ```
   $ drweb-ctl update
   ```

If the error persists, contact our technical support and provide the error code.

Error message: *Kernel module is not available*

Error code: x113

Internally used name: EC_NO_KERNEL_MODULE

Description: The Linux kernel module required for the operation of SpIDer Guard is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Check which operating mode of the component was selected and change it, if necessary, by setting the value to *Auto* (for the Mode parameter in the [LinuxSpider] section of the configuration file).

   - Use commands of the command-line management tool.

To set the value to *Auto*, run the command:

```
# drweb-ctl cfset LinuxSpider.Mode Auto
```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset LinuxSpider.Mode -r
```

2. If the error persists, build and install the loadable kernel module for SpIDer Guard manually.

> Operation of SpIDer Guard and of the loadable kernel module is guaranteed only if your OS is on the list of supported systems (see the System Requirements and Compatibility section).

If the error persists, contact our technical support and provide the error code.

Error message: *MeshD is not available*

Error code: `x114`

Internally used name: `EC_NO_MESHD`

Description: The Dr.Web MeshD component required for load balancing during the scanning of files is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-meshd` executable file. Change the path if necessary (the `ExePath` parameter in the `[MeshD]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

```
$ drweb-ctl cfshow MeshD.ExePath
```

   To set a new parameter value, run the command:

```
# drweb-ctl cfset MeshD.ExePath <new path>
```

   To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset MeshD.ExePath -r
```

2. If the configuration does not contain Dr.Web MeshD settings or the error persists after entering the correct path, install or reinstall the `drweb-meshd` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *UrlCheck is not available*

Error code: `x116`

Internally used name: `EC_NO_URL_CHECK`

Description: The Dr.Web URL Checker component required for checking URLs for belonging to blocked or potentially dangerous categories is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-urlcheck` executable file. Change the path if necessary (the `ExePath` parameter in the `[URLCheck]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow URLCheck.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset URLCheck.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset URLCheck.ExePath -r
   ```

2. If the configuration does not contain Dr.Web URL Checker settings or the error persists after entering the correct path, install or reinstall the `drweb-urlcheck` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *GateD is not available*

Error code: `x117`

Internally used name: `EC_NO_GATED`

Description: The SpIDer Gate component required for scanning network connections is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-gated` executable file. Change the path if necessary (the `ExePath` parameter in the `[GateD]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow GateD.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset GateD.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset GateD.ExePath -r
   ```

2. If the configuration does not contain SpIDer Gate settings or the error persists after entering the correct path, install or reinstall the `drweb-gated` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *MailD is not available*

Error code: `x118`

Internally used name: `EC_NO_MAILD`

Description: The Dr.Web MailD component required for email scanning is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-maild` executable file. Change the path if necessary (the `ExePath` parameter in the `[MailD]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow MailD.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset MailD.ExePath <new path>
   ```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset MailD.ExePath -r
```

2. If the configuration does not contain Dr.Web MailD settings or the error persists after entering the correct path, install or reinstall the `drweb-maild` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *ScanEngine is not available*

Error code: `x119`

Internally used name: `EC_NO_SCAN_ENGINE`

Description: The Dr.Web Scanning Engine component required for threat detection is missing or has failed to start.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-se` executable file. Change the path if necessary (the `ExePath` parameter in the `[ScanEngine]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow ScanEngine.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset ScanEngine.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset ScanEngine.ExePath -r
   ```

2. If the error persists after entering the correct path:

   - Run the command:

     ```
     $ drweb-ctl rawscan /
     ```

   If the line “`Error: No valid license provided`” is output, a valid key file is missing. Register Dr.Web Security Space and receive a license. After receiving the license, check whether the key file is available and install it if necessary.

   - If your operating system uses SELinux, configure the security policy for the `drweb-se` module (see the section Configuring SELinux Security Policies).

3. If the configuration does not contain Dr.Web Scanning Engine settings or the previous steps did not help, install or reinstall the `drweb-se` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *FileCheck is not available*

Error code: `x120`

Internally used name: `EC_NO_FILE_CHECK`

Description: The Dr.Web File Checker component is missing or has failed to start.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-filecheck` executable file. Change the path if necessary (the `ExePath` parameter in the `[FileCheck]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow FileCheck.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset FileCheck.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset FileCheck.ExePath -r
   ```

2. If the error persists after entering the correct path:

   - If your operating system uses SELinux, configure the security policy for the `drweb-filecheck` module (see the section Configuring SELinux Security Policies).

3. If the configuration does not contain Dr.Web File Checker settings or the previous steps did not help, install or reinstall the `drweb-filecheck` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *ESAgent is not available*

Error code: `x121`

Internally used name: `EC_NO_ESAGENT`

Description: The Dr.Web ES Agent component required to connect to a centralized protection server is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-esagent` executable file. Change the path if necessary (the `ExePath` parameter in the `[ESAgent]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow ESAgent.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset ESAgent.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset ESAgent.ExePath -r
   ```

2. If the configuration does not contain Dr.Web ES Agent settings or the error persists after entering the correct path, install or reinstall the `drweb-esagent` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Firewall is not available*

Error code: `x122`

Internally used name: `EC_NO_FIREWALL`

Description: The Dr.Web Firewall for Linux component required for scanning network connections is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-firewall` executable file. Change the path if necessary (the `ExePath` parameter in the `[LinuxFirewall]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow LinuxFirewall.ExePath
   ```

To set a new parameter value, run the command:

```
# drweb-ctl cfset LinuxFirewall.ExePath <new path>
```

To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset LinuxFirewall.ExePath -r
```

2. If the configuration does not contain Dr.Web Firewall for Linux settings or the error persists after entering the correct path, install or reinstall the `drweb-firewall` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *NetCheck is not available*

Error code: `x123`

Internally used name: `EC_NO_NETCHECK`

Description: The Dr.Web Network Checker component required to scan files over the network is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the `/var/log/syslog` file or the `/var/log/messages` file, depending on the OS). You can also use the `drweb-ctl log` command.

Troubleshooting

1. Check the path to the `drweb-netcheck` executable file. Change the path if necessary (the `ExePath` parameter in the `[Netcheck]` section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

```
$ drweb-ctl cfshow Netcheck.ExePath
```

   To set a new parameter value, run the command:

```
# drweb-ctl cfset Netcheck.ExePath <new path>
```

   To reset the parameter to its default value, run the command:

```
# drweb-ctl cfset Netcheck.ExePath -r
```

2. If the configuration does not contain Dr.Web Network Checker settings or the error persists after entering the correct path, install or reinstall the `drweb-netcheck` package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *CloudD is not available*

Error code: x124

Internally used name: EC_NO_CLOUDD

Description: The Dr.Web CloudD component required for making requests to the Dr.Web Cloud service is missing.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

1. Check the path to the drweb-cloudd executable file. Change the path if necessary (the ExePath parameter in the [CloudD] section of the configuration file).

   You can also use the commands of the command-line management tool.

   To get the current parameter value, run the command:

   ```
   $ drweb-ctl cfshow CloudD.ExePath
   ```

   To set a new parameter value, run the command:

   ```
   # drweb-ctl cfset CloudD.ExePath <new path>
   ```

   To reset the parameter to its default value, run the command:

   ```
   # drweb-ctl cfset CloudD.ExePath -r
   ```

2. If the configuration does not contain Dr.Web CloudD settings or the error persists after entering the correct path, install or reinstall the drweb-cloudd package.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support and provide the error code.

Error message: *Unexpected error*

Error code: x125

Internally used name: EC_UNEXPECTED_ERROR

Description: An unexpected error has occurred in operation of one or several components.

To identify a possible cause and circumstances of the error, refer to the Dr.Web Security Space log (by default, it is located in the /var/log/syslog file or the /var/log/messages file, depending on the OS). You can also use the drweb-ctl log command.

Troubleshooting

Restart Dr.Web Security Space by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support and provide the error code.

Errors Without Code

Symptoms: After installation of the SpIDer Guard kernel module, the operating system abnormally shuts down with the "*Kernel panic*" error.

Description: The SpIDer Guard kernel module cannot operate in operating system kernel environment (for example, when the OS operates in the Xen hypervisor environment).

Troubleshooting

1. Cancel loading of the SpIDer Guard kernel module (named `drweb`) by adding the following string in the GNU GRUB boot loader:

   ```
   drweb.blacklist=yes
   ```

   to the string of kernel boot options.
2. When the OS is loaded, remove the `drweb.ko` module from the directory `/lib/modules/`uname -r`/extra` storing additional modules.
3. Set the operation mode of SpIDer Guard to *Auto* by running the commands:

   ```
   # drweb-ctl cfset LinuxSpider.Mode Auto
   # drweb-ctl reload
   ```

4. If your operating system does not support *fanotify* or this mode does not allow using SpIDer Guard to fully control the file system, and using the *LKM* mode becomes obligatory, do not use the Xen hypervisor.

If the error persists, contact our technical support.

Symptoms: Such components as SpIDer Gate, Dr.Web MailD do not scan messages; Dr.Web Security Space log indicates `Too many open files`.

Description: Owing to a heavy load while scanning data, Dr.Web Network Checker has reached the limit on the number of available file descriptors.

Troubleshooting

1. Raise the limit of the number of open file descriptors available to the application by running the `ulimit -n` command (the default limit on the number of descriptors for Dr.Web Security Space is 16384).

The systemd system service can ignore the specified limit changes in some cases.

In these situations, edit the file /etc/systemd/system/drweb-configd.service.d/limits.conf (or create it if it does not exist) and specify the changed limit value, for example:

```
[Service]
LimitNOFILE=16384
```

Available systemd limits can be determined using the man systemd.exec command.

2. Once the limit is changed, restart Dr.Web Security Space by running the command:

```
# service drweb-configd restart
```

If the error persists, contact our technical support.

Symptoms: The main window of Dr.Web Security Space is disabled, the status indicator in the notification area of the taskbar displays a critical error mark, and the drop-down list contains only one disabled item—**Loading**.

Description: Dr.Web Security Space cannot start as the Dr.Web ConfigD configuration daemon is not available.

Troubleshooting

1. Run the command:

```
# service drweb-configd restart
```

to restart Dr.Web ConfigD and Dr.Web Security Space in general.

2. If this command returns an error or has no effect, install the drweb-configd package separately.

This also may mean that PAM authentication is not used in the system. If so, install and configure PAM since Dr.Web Security Space cannot operate correctly without it.

3. If the error persists, uninstall Dr.Web Security Space entirely and then reinstall it.

For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support.

Symptoms

1. The status indicator is not displayed in the notification area after the user has logged in.
2. Running the command to start the graphical interface:

   ```
   $ drweb-gui
   ```

   opens Dr.Web Security Space main window.

Description: Possibly, this issue is caused by that the required additional library `libappindicator1` is not installed in your system.

Troubleshooting

1. Make sure that the `libappindicator1` package is installed in your system by running the command:

   ```
   # dpkg -l | grep libappindicator1
   ```

2. If the command does not output any result to the screen, you should install the package using any available system package manager. After that, log out and then log in again.

   Note that this may also mean that PAM is not used in the system for user authentication. If so, install and configure it.
3. If the previous actions did not help, uninstall Dr.Web Security Space completely and then install it again.

   For details on how to install and uninstall Dr.Web Security Space or its components, refer to sections Installing Dr.Web Security Space and Uninstalling Dr.Web Security Space.

If the error persists, contact our technical support.

Symptoms

1. After disabling SpIDer Gate, all network connections (both outgoing and, possibly, incoming via the SSH and FTP) stop working.
2. Searching through NetFilter (`iptables`) rules using the command:

   ```
   # iptables-save | grep "comment --comment --comment"
   ```

   returns a non-empty result.

Description: This error is related to incorrect operation of NetFilter (`iptables`) earlier than 1.4.15. The error causes the rules having a unique tag (comment) to be added incorrectly, as a result of which, when shutting down, SpIDer Gate cannot remove its rules for forwarding network connections.

Troubleshooting

1. Re-enable SpIDer Gate.

2. If you need to keep SpIDer Gate disabled, remove incorrect NetFilter (iptables) rules by running the command:

   ```
   # iptables-save | grep -v "comment --comment --comment" | iptables-
   restore
   ```

   Running the iptables-save and iptables-restore commands requires superuser privileges. To get superuser privileges, use the su or sudo command.

   This command removes all rules with an incorrectly added comment from the list of rules (for example, those that were added by other applications adjusting traffic routing).

Additional Information

- To prevent this issue in the future, it is recommended to upgrade your operating system (or at least NetFilter to version 1.4.15 or later).
- You can also switch SpIDer Gate to a manual mode of forwarding connections by specifying the required rules manually using the iptables utility (not recommended).
- For details, refer to the following man pages: drweb-firewall(1), drweb-gated(1), iptables(8).

If the error persists, contact our technical support.

Symptoms: Double-clicking an icon of a file or a directory in a graphical file manager starts the scanning in Dr.Web Security Space instead of opening this file or directory.

Description: The graphical shell has automatically associated files of a certaing type and/or directories with the **Open With Dr.Web Security Space** action.

Troubleshooting

1. Cancel the association of files of this type with Dr.Web Security Space. The associations are registered in the mimeapps.list or defaults.list file. Files defining local settings changed in a user profile are stored in the ~/.local/share/applications/ or ~/.config/ directory (these directories usually have a “hidden” attribute).
2. Open the mimeapps.list or defaults.list file with any text editor. Note that, to edit the system association file, superuser privileges are required. If necessary, use the su or sudo command.
3. In the file, find the [Default Applications] section and association lines such as <*MIME-type*>=drweb-gui.desktop, for example:

   ```
   [Default Applications]
   inode/directory=drweb-gui.desktop
   text/plain=drweb-gui.desktop;gedit.desktop
   ```

4. If the right part of the association line (after the equal sign) contains links to other applications besides drweb-gui.desktop, remove only the link to the drweb-gui application (drweb-

gui.desktop). If the association contains a link only to the drweb-gui application, remove the whole association line.

5. Save the changed file.

Additional information

- To check the current associations, use the xdg-mime, xdg-open and xdg-settings utilities (included in the xdg-utils package).
- To read more about the xdg utilities, refer to the documentation shown by running the man command: xdg-mime(1), xdg-open(1) and xdg-settings(1).

If the error persists, contact our technical support.

10.8. Appendix H. List of Abbreviations

The following abbreviations were used in this manual without further interpretation:

Convention	Complete form
AD	Microsoft Active Directory
FQDN	Fully Qualified Domain Name
GID	Group ID (system user group identifier)
GNU	GNU project (GNU is Not Unix)
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure (via SSL/TLS)
ID	Identifier
IMA/EVM	Integrity Measurement Architecture and Extended Verification Module
IP	Internet Protocol
LKM	Loadable Kernel Module
MBR	Master Boot Record
OS	Operating System
PID	Process ID (system process identifier)
PAM	Pluggable Authentication Modules
RPM	Red Hat Package Manager
SP	Service Pack
SSH	Secure Shell
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UID	User ID (system user identifier)
URI	Uniform Resource Identifier

Convention	Complete form
URL	Uniform Resource Locator
VBR	Volume Boot Record