kaspersky

Kaspersky Embedded Systems Security for Linux

© 2026 AO Kaspersky Lab

Contents

Kaspersky Embedded Systems Security 4.0 for Linux Help

Kaspersky Embedded Systems Security 4.0 for Linux

- Distribution kit
- Hardware and software requirements
  - Hardware requirements
  - Software requirements
  - Special considerations involved with using the application in different operating systems
  - Supported versions of Kaspersky Security Center

What's new

Preparing to install Kaspersky Embedded Systems Security

Installation and initial configuration of Kaspersky Embedded Systems Security

- The installation and initial configuration of Kaspersky Security Center Network Agent
- Installing the Kaspersky Embedded Systems Security management plug-ins
  - Installing the Kaspersky Embedded Systems Security web plug-in
  - Installing the Kaspersky Embedded Systems Security MMC plug-in
- Installing and initially configuring the application using Kaspersky Security Center
  - Creating an installation package in the Web Console
  - Initial configuration settings in the Web Console
  - Creating an installation package in the Administration Console
  - Initial configuration settings in the Administration Console
  - Preparing an archive with application databases in order to create an installation package with integrated databases
  - Autoinstall.ini configuration file settings
  - Getting started using Kaspersky Security Center
- Installing and initially configuring the application using the command line
  - Installing the application using the command line
  - Post-installation configuration of the application in interactive mode
    - Selecting the locale
    - Viewing the End User License Agreement and the Privacy Policy
    - Accepting the End User License Agreement
    - Accepting the Privacy Policy
    - Using Kaspersky Security Network
    - Removing users from privileged groups
    - Assigning the Administrator role to a user
    - Determining the file operation interceptor type
    - Enabling automatic configuration of SELinux
    - Configuring the update source
    - Configuring proxy server settings
    - Starting an application database update
    - Enabling automatic application database update
    - Activating the application
  - Post-installation configuration of the application in automatic mode
  - Settings in the configuration file for post-installation configuration
- Configuring allowing rules in the SELinux system

Running the application on Astra Linux OS in closed software environment mode
Running the application on ALT SP in closed software environment mode
Optimizing resource consumption by the application
Recommendations for typical server roles
Analysis of application performance on the device
Analyzing the resource consumption of the File Threat Protection task
Identifying the most frequently scanned objects
Recommendations for File Threat Protection and scan tasks
Configuring scan exclusions
Configuring exclusions from file operation interception
Setting the application memory usage limit
Recommendations for the Web Threat Protection and Network Threat Protection tasks
Updating the application from a previous version
About updating Kaspersky Embedded Systems Security management plug-ins
Updating the application using Kaspersky Security Center
Updating the application using the command line
Special considerations when setting parameter values when updating the application
Uninstalling the application
Uninstalling the application and Network Agent using Kaspersky Security Center
Uninstalling the application using the command line
Network Agent removal using the command line
Removing Kaspersky Embedded Systems Security management plug-ins
Application licensing
About the End User License Agreement
About the license
About license certificate
About license key
About activation code
About the key file
About subscription
Application activation and license key management
Activating the application using Kaspersky Security Center
License key management on the command line
Viewing information about used license keys
Viewing information about license keys on a device using the Web Console
Viewing information about license keys on a device using the Administration Console
Viewing information about the license and the key in the command line
Data provision
Data provided when using an activation code
Data provided when downloading updates from Kaspersky update servers
Data sent to Kaspersky Security Center
Data provided when following links in the application interface
Data provided when using Kaspersky Security Network
Application management concept
Managing the application using Kaspersky Security Center
About Kaspersky Embedded Systems Security management plug-ins
Kaspersky Security Center policies

Tasks for Kaspersky Embedded Systems Security created in Kaspersky Security Center
Logging in and out of the Web Console and Cloud Console
Managing policies in the Web Console
Creating a policy in the Web Console
Changing policy settings in the Web Console
Policy settings in the Web Console
Managing policies in the Administration Console
Creating a policy using the Administration Console
Changing policy settings in the Kaspersky Security Center Administration Console
Policy settings in the Administration Console
Managing tasks in the Web Console
Creating tasks in the Web Console
Changing task settings in the Web Console
Starting, stopping, pausing, and resuming tasks in the Web Console
Managing tasks in the Administration Console
Creating tasks in the Administration Console
Changing task settings in the Administration Console
Starting, stopping, pausing, and resuming tasks in the Administration Console
Managing the application using the command line
Enabling automatic addition of kess-control commands (bash completion)
Task management on the command line
Viewing a list of tasks in the command line
Viewing the status of a task in the command line
Creating a task in the command line
Starting, stopping, pausing, and resuming tasks in the command line
Deleting a task in the command line
Displaying task settings in the command line
Editing task settings in the command line
Editing task settings using a configuration file
Editing task settings using the command line options
Restoring default task settings in the command line
Configuring task schedule in the command line
Managing general application settings on the command line
Displaying general application settings
Editing general application settings
Using filters to limit results of queries
Exporting and importing application settings
Managing user roles using the command line
Viewing a list of users and roles
Assigning a role to a user
Revoking a user role
Starting and stopping the application
Starting and stopping the application using the Web Console
Starting and stopping the application using the Administration Console
Starting and stopping the application using the command line

Viewing device protection status and application performance information
Viewing the protection status of a device in the Web Console
Viewing the protection status of a device in the Administration Console
Viewing information about the operation of an application in the Web Console
Viewing information about the operation of an application in the Administration Console
Viewing information about the operation of an application in the command line
Viewing application statistics
Viewing application statistics in the Web Console
Viewing application statistics in the Administration Console
Viewing a list of mount points in the Web Console
Viewing the list of mount points in the Administration Console
Viewing application statistics and the list of mount points in the command line
Obtaining system performance metrics
Updating application databases and modules
Updating databases and modules
Updating sources and update scenarios
Updating application databases and modules in the Web Console
Updating application databases and modules in the Administration Console
Updating application databases and modules in the command line
Updating using Kaspersky Update Utility
Rolling back application database and module updates
File Threat Protection
Configuring File Threat Protection in the Web Console
Configuring File Threat Protection in the Administration Console
Scan scopes window
<New scan scope> window
Scan settings window
Actions on threat detection window
File Threat Protection exclusions
Exclusion scopes window
<New exclusion scope> window
Exclusions by mask window
Exclusions by threat name window
Exclusions by process window
Trusted process window
Configuring File Threat Protection on the command line
File Threat Protection task settings
Optimizing network directory scanning
Special considerations for scanning symbolic links and hard links
Malware Scan
Malware Scan in the Web Console
Add scan scope window
Scan scopes section
Scan scopes window
Exclusion scopes section
Exclusion scopes window
Add exclusion scope window

Exclusions by mask window
Exclusions by threat name window
Malware Scan in the Administration Console
Scan scopes window
<New scan scope> window
Scan scope settings window
Scan scopes window
Scan settings window
Actions on threat detection window
Exclusions section
Exclusion scopes window
<New exclusion scope> window
Exclusions by mask window
Exclusions by threat name window
Malware Scan in the command line
Settings of the Malware Scan predefined task
Custom Scan of files and directories on the command line
Critical Areas Scan
Critical Areas Scan in the Web Console
Add scan scope window
Scan scopes section
Scan scopes window
Exclusion scopes section
Exclusion scopes window
Add exclusion scope window
Exclusions by mask window
Exclusions by threat name window
Critical Areas Scan in the Administration Console
Scan scopes window
<New scan scope> window
Scan scope settings window
Scan scopes window
Scan settings window
Actions on threat detection window
Exclusions section
Exclusion scopes window
<New exclusion scope> window
Exclusions by mask window
Exclusions by threat name window
Critical Areas Scan on the command line
Removable drives scan
Configuring Removable Drives Scan in the Web Console
Configuring Removable Drives Scan in the Administration Console
Configuring Removable Drives Scan on the command line
Firewall Management
About dynamic rules
About network packet rules

- About the predefined available networks
- Firewall Management in the Web Console
- Firewall Management in the Administration Console
  - Network packet rules window
  - Added network packet rule window
  - Add IP addresses window
  - Available networks window
  - Network connection window
- Firewall Management in the command line
  - Configuring a list of network packet rules on the command line
  - Configuring network zones on the command line

Web Threat Protection
- Configuring Web Threat Protection in the Web Console
- Configuring Web Threat Protection in the Administration Console
  - Trusted web addresses window
  - Scan settings window
- Configuring Web Threat Protection on the command line

Scanning network traffic
- About network traffic interception modes
- Configuring network traffic scanning in the Web Console
- Configuring network traffic scanning in the Administration Console
  - Trusted domains window
  - Trusted certificates window
  - Traffic interception exclusions window
  - Traffic interception exclusion window
  - Monitored ports
- Configuring network traffic scanning on the command line
  - Viewing and editing network traffic scan settings
  - Viewing exclusions from encrypted connections scan
  - Managing the list of trusted root certificates
  - Configuring and viewing traffic interception exclusions
- Exclusions from traffic interception using the tools of the operating system

Network Threat Protection
- Configuring Network Threat Protection in the Web Console
- Configuring Network Threat Protection in the Administration Console
  - Exclusions window
- Configuring Network Threat Protection on the command line

BadUSB Attack Prevention
- Configuring BadUSB Attack Prevention in the Web Console
- Configuring BadUSB Attack Prevention in the Administration Console
- Configuring BadUSB Attack Prevention on the command line

Protection against remote malicious encryption
- Configuring Anti-Cryptor in the Web Console
- Configuring Anti-Cryptor in the Administration Console
  - Scan scopes window
  - <New scan scope> window
  - Protection settings window

- Exclusion scopes window
- <New exclusion scope> window
- Exclusions by mask window
- Configuring Anti-Cryptor on the command line

Managing blocked devices

Application Control

- About Application Control rules
- Configuring Application Control in the Web Console
- Configuring Application Control in the Administration Console
  - Application Control rules window
  - Adding rule window
  - Application categories window
  - User or group window
  - Trusted certificates of Application Control window
- Configuring Application Control on the command line
  - Application Control task settings
  - Creating and editing a list of categories
  - Viewing the list of categories
  - Configuring the Application Control rule list
  - Managing the list of trusted certificates of Application Control
  - Exporting and importing Application Control settings

Inventory

- Configuring Inventory in the Web Console
  - Add scan scope window
  - Exclusion scopes section
  - Exclusion scopes window
  - Add exclusion scope window
- Configuring Inventory in the Administration Console
  - Scan scopes window
  - <New scan scope> window
  - Exclusions section
  - Exclusion scopes window
  - <New exclusion scope> window
- Configuring Inventory on the command line
  - Inventory task settings
  - Viewing a list of detected applications

Device Control

- Configuring Device Control in the Web Console
- Configuring Device Control in the Administration Console
  - Trusted devices window
  - Trusted device window
  - Device window on client devices
  - List of devices based on Device Control events window
  - Access to trusted device window
  - User or group window
  - Device type window
  - Configure device access rule window

- User or group window
- Access schedule window
- Connection buses window
- Configuring the Device Control message template
- Configuring Device Control on the command line
- Device Control task settings
- Viewing and exporting the list of connected devices on the command line
- Managing the list of trusted devices on the command line
- Granting temporary access to device
- Granting access to a blocked device in the Web Console
- Gaining temporary access to the device on the command line

System Integrity Monitoring

- Real-time System Integrity Monitoring
- Configuring System Integrity Monitoring in the Web Console
- Configuring System Integrity Monitoring in the Administration Console
- Scan scopes window
- <New scan scope> window
- Exclusion scopes window
- <New exclusion scope> window
- Exclusions by mask window
- Configuring System Integrity Monitoring on the command line
- System Integrity Check
- Configuring System Integrity Check in the Web Console
- Add scan scope window
- Exclusion scopes section
- Exclusion scopes window
- Add exclusion scope window
- Exclusions by mask window
- Configuring System Integrity Check in the Administration Console
- Scan scopes window
- <New scan scope> window
- Exclusion scopes section
- Exclusion scopes window
- <New exclusion scope> window
- Exclusions by mask window
- Configuring System Integrity Check on the command line

Behavior Detection

- Configuring Behavior Detection in the Web Console
- Configuring Behavior Detection in the Administration Console
- Exclusions by process window
- Trusted process window
- Configuring Behavior Detection on the command line

Using Kaspersky Security Network
Configuring the use of Kaspersky Security Network in the Web Console
Configuring the use of Kaspersky Security Network in the Administration Console
Kaspersky Security Network settings
Kaspersky Security Network Statement
Kaspersky Private Security Network Statement
Configuring the use of Kaspersky Security Network on the command line
Checking the connection to Kaspersky Security Network using the command line
Enabling and disabling cloud mode from the command line
Advanced application settings
Configuring a proxy server
Configuring proxy server settings in the Web Console
Configuring proxy server settings in the Administration Console
Configuring proxy server settings on the command line
Configuring global exclusions
Configuring global exclusions in the Web Console
Configuring global exclusions in the Administration Console
Mount point path window
Configuring global exclusions on the command line
Excluding process memory from scans
Selecting the interception mode for file operations
Detection of applications that hackers can use to harm
Enabling application stability monitoring
Configuring application startup settings
Limiting the use of resident memory by the application
Limiting the use of memory and processor resources
Limiting the number of Custom Scan tasks
Configuring the transfer of data to Kaspersky Security Center storage
Configuring permissions for task management
Enabling or disabling monitoring of namespaces
Canceling tasks when running on battery power
Managing additional telemetry settings
Trusted zone
Backup
Configuring Backup settings in the Web Console
Configuring Backup settings in the Administration Console
Configuring Backup settings on the command line
Working with Backup objects on the command line
Integration with Kaspersky Managed Detection and Response
About response actions
Uploading the KPSN configuration file for integration with Kaspersky Managed Detection and Response
Configuring the Kaspersky Managed Detection and Response integration in the Web Console
Configuring the Kaspersky Managed Detection and Response integration in the Administration Console
Configuring the Kaspersky Managed Detection and Response integration on the command line
Network isolation of a device

- Managing the Quarantine
  - Editing Quarantine settings in the Web Console
  - Editing Quarantine settings in the Administration Console
  - Editing Quarantine settings on the command line
  - Managing quarantined files on the command line
- Viewing events and reports
  - Configuring event logging to the operating system log
  - Configuring application event log settings
  - Viewing events in Kaspersky Security Center
  - Viewing events on the command line
  - Configuring pop-up notifications in the graphical interface
- Application management via the graphical interface
  - Graphical user interface
  - Enabling and disabling application components
  - Starting and stopping scan tasks
  - Starting a custom scan
  - Starting and stopping the Update task
  - Configuring Kaspersky Security Network
  - Viewing reports
  - Viewing Backup objects
  - Managing license keys
    - Adding a license key
    - Removing a license key
    - Viewing license information
  - Gaining temporary access to device
  - Viewing information about the application
  - Creating a trace file
- Application components integrity check
- Contact Technical Support
  - Technical Support via Kaspersky CompanyAccount
  - Obtaining information for Technical Support
    - About application trace files
    - Configuring application trace settings
    - Application management plug-in trace files
    - About dump files
    - Managing dump file settings
  - Remote device diagnostics using Kaspersky Security Center
- Appendices
  - Appendix 1. Commands for managing the application
    - Commands for managing application tasks and settings
      - Commands for managing general application settings
      - Commands for managing task settings
      - Commands for managing tasks
      - Commands for managing network traffic scan settings
    - Statistics commands
    - Commands for displaying events
    - Commands for managing application events

Commands for managing license keys
Commands for Firewall Management
Commands used to manage blocked devices
Commands for managing Device Control
Commands for managing Application Control
Commands for managing Backup
Commands for managing the Quarantine
Commands for managing device network isolation
Commands for managing users and roles
Commands for managing system performance metrics
Appendix 2. Configuration files and default application settings
Rules for editing task configuration files
Preset configuration files
Default settings for command line tasks
Default settings for the File_Threat_Protection task (ID:1)
Default settings for the Scan_My_Computer task (ID:2)
Default settings for the Scan_File task (ID:3)
Default settings for the Critical_Areas_Scan task (ID:4)
Default settings for the Update task (ID:6)
Default settings for the System_Integrity_Monitoring task (ID:11)
Default settings for the Firewall_Management task (ID:12)
Default settings for the Anti_Cryptor task (ID:13)
Default settings for the Web_Threat_Protection task (ID:14)
Default settings for the Device_Control task (ID:15)
Default settings for the Removable_Drives_Scan task (ID:16)
Default settings for the Network_Threat_Protection task (ID:17)
Default settings for the Behavior_Detection task (ID:20)
Default settings for the Application_Control task (ID:21)
Default settings for the Inventory_Scan task (ID:22)
Default settings of the BadUSB Attack Prevention task (ID: 37)
General application settings
Network traffic scan settings
Tasks schedule settings
Appendix 3. Command line return codes
Appendix 4. Settings for interaction with external services
Sources of information about Kaspersky Embedded Systems Security
Glossary
Activating the application
Active key
Active policy
Administration group
Administration Server
Application databases
Application settings
Database of malicious web addresses
Database of phishing web addresses
Disinfection

Exclusion
False positive
File mask
Group policy
Group task
Infected object
Kaspersky update servers
License
License certificate
Policy
Proxy server
Reserve key
Startup objects
Subscription
Trusted device
Information about third-party code
Trademark notices

Kaspersky Embedded Systems Security 4.0 for Linux Help

	What's new Learn what's new in this version of the application.		**Updating application databases and modules** Check the performance of the protection system.
	Hardware and software requirements Check the hardware requirements and supported operating systems.		**Resource consumption optimization** Learn how to optimize the usage of operating system resources.
	Installation and initial configuration Learn how to prepare the device for application installation, how to install the application, and complete the initial configuration of the application.		**Monitoring & Reporting** Check the protection status of your device and learn how to view events and reports of the application.
	Upgrading a previous version of the application Learn how to upgrade Kaspersky Embedded Systems Security for Linux to version 4.0.		**Data provision** Learn about data provision and the protection of personal data and confidential information.
	Licensing and activation Get to know the licensing terminology and learn about application activation.		**Contacting Technical Support** Learn about how to contact Technical Support.

Kaspersky Embedded Systems Security 4.0 for Linux

Kaspersky Embedded Systems Security 4.0 for Linux ("Kaspersky Embedded Systems Security," "Application") designed to protect devices running Linux® operating systems against various types of threats, including network and scam attacks.

The application allows you to protect both physical devices and virtual machines.

The application is not intended for industrial processes that use automated control systems. To protect devices in such systems, we recommend using Kaspersky Industrial CyberSecurity for Linux Nodes.

The following functional components and tasks of the application provide the main functions of device protection and control:

- **File Threat Protection** prevents infection of the file system on the user device. The *File Threat Protection* component starts automatically when Kaspersky Embedded Systems Security is launched and scans all files that are opened, saved, and started in real time.

  You can also scan protected devices on demand using the following scan tasks:

  - **Malware Scan**. The application scans for the presence of malware in file system objects located on local disks of the device, as well as mounted and shared resources, which are accessed via SMB and NFS protocols. You can use this task to perform a full or custom scan of the device.

  - **Critical Areas Scan**. The application scans boot sectors, startup objects, process memory, and kernel memory.

- **Removable Drives Scan**. The *Removable Drives Scan* component allows you to monitor the connection of removable drives to the device in real time and scan a removable drive and its boot sectors for malware. Kaspersky Embedded Systems Security can scan the following removable drives: CDs, DVDs, Blu-ray discs, flash drives (including USB modems), external hard drives, and floppy disks.

- **Web Threat Protection**. The *Web Threat Protection* component allows you to scan inbound traffic, prevent downloads of malicious files from the Internet, and block phishing, adware, and other malicious websites. Kaspersky Embedded Systems Security can scan encrypted connections.

- **Network Threat Protection**. The *Network Threat Protection* component allows you to scan inbound network traffic for activity that is typical for network attacks.

- **Firewall Management**. The *Firewall Management* component allows you to monitor the firewall settings of the operating system and filter all network activity in accordance with the network packet rules that you have configured.

- **BadUSB Attack Prevention**. The *BadUSB Attack Prevention* protection component allows preventing the connection of infected USB devices that imitate a keyboard to the computer.

- **Anti-Cryptor**. The *Anti-Cryptor* component allows you to scan remote devices' calls to files located in local directories with network access via SMB/NFS protocols and protect files from remote malicious encryption.

- **Device Control**. The *Device Control* component allows you to manage user access to the devices that are installed on or connected to the client device (for example, hard drives, cameras, or Wi-Fi modules). This lets you protect the client device from infection when external devices are connected, and prevent data loss or leaks. User access to devices is governed by access regimes and access rules that you have configured.

- **Application Control**. The *Application Control* component allows you to manage the launch of applications on user devices. This reduces the risk of device infection by restricting access to applications. Application launching is regulated by the Application Control rules that you have configured.

- **Inventory**. The *Inventory* task provides information about all applications executable files stored on the client devices. This information can be useful, for example, for creating Application Control rules.

- **Behavior Detection**. The *Behavior Detection* component allows you to monitor for any malicious activity from applications in the operating system. When malicious activity is detected, Kaspersky Embedded Systems Security can terminate the process of the application that performs malicious activity.

- **System Integrity Monitoring** allows you to track changes to files and directories of the operating system. The *System Integrity Monitoring* component monitors the actions performed with objects from the monitoring scope specified in the component settings in real time. You can use the *System Integrity Check* task to check the integrity of the system on demand. The check is performed by comparing the current states of objects included in the monitoring scope with their initial states, which were previously established as a baseline.

Kaspersky Embedded Systems Security allows you to detect infected objects and neutralize the threats detected in them. For this, the application can use:

- Application databases to detect and disinfect infected files. During the scan process, the application analyzes each file for the presence of a threat: it compares the file code with the code of a specific threat and looks for possible matches.

- Kaspersky Security Network. The use of data from Kaspersky Security Network ensures faster responses by Kaspersky Embedded Systems Security to various threats, improves the performance of some protection components, and reduces the likelihood of false positives.

Prior to disinfection or removal, Kaspersky Embedded Systems Security saves backup copies of files in the Backup located on the device. If after disinfection, you partially or completely lose access to important information in a disinfected file, you can restore the file from the copy.

While performing scan tasks, Kaspersky Embedded Systems Security can disinfect and delete files that are protected from modification: files with the 'immutable' and 'append-only' attributes and files in directories with the 'immutable' and 'append-only' attributes. Backup stores copies of these files that were created before disinfection or deletion. You can restore files from backup copies, if necessary. When scan tasks are completed, the 'immutable' and 'append-only' attributes of disinfected files are not preserved.

Kaspersky Embedded Systems Security can operate in Notify-only mode. *Notify-only mode* is an operation mode for the application in which, if a threat is detected, application components and tasks do not attempt to disinfect or delete malicious objects, deny access or block the activity of applications. Instead, the application only informs the user about the detected threat.

Kaspersky Embedded Systems Security supports integration with the Kaspersky Managed Detection and Response solution to extend the functionality of the application. Integration with Kaspersky Managed Detection and Response enables continuous search, detection and elimination of threats aimed at your organization.

To keep the application up to date, additional application functions are provided:

- Activating the application with a key file or activation code.
- Updating the databases and application modules from Kaspersky update servers, via the Administration Server, or from a user-specified source on schedule and on demand.
- User access control for the application functions according to the user roles.
- Notification of the administrator about events that occurred while the application was running.
- Integrity check of application components using the integrity check tool.

You can manage Kaspersky Embedded Systems Security using the following methods:

- Using Kaspersky Security Center through the Kaspersky Security Center Web Console, Kaspersky Security Center Cloud Console, or the Administration Console.
- Using control commands from the command line.
- Using a graphical user interface.

The update functionality (including anti-virus signature updates and code base updates), as well as the KSN functionality may not be available in the application in the territory of the USA.

Distribution kit

You can download the files that are included in the Kaspersky Embedded Systems Security distribution kit, as well as the files needed to remotely install the application using Kaspersky Security Center, on the Kaspersky website.

The Kaspersky Embedded Systems Security distribution kit includes the following files:

- kess-4.0.0-<build number>.i386.rpm, kess_4.0.0-<build number>_i386.deb

  These contain the main application files. Packages can be installed to 32-bit operating systems based on the type of package manager.

- kess-4.0.0-<build number>.x86_64.rpm, kess_4.0.0-<build number>_amd64.deb

  These contain the main application files. Packages can be installed to 64-bit operating systems based on the type of package manager.

- kess_4.0.0-<build number>.aarch64.rpm, kess_4.0.0-<build number>_arm64.deb

  These contain the main application files. Packages for the relevant package manager can be installed on 64-bit operating systems for the Arm® architecture.

- kess-gui-4.0.0-<build number>.i386.rpm, kess-gui-4.0.0-<build number>_i386.deb

  These contain the files of the application graphical user interface. Packages can be installed to 32-bit operating systems based on the type of package manager.

- kess-gui-4.0.0-<build number>.x86_64.rpm, kess-gui-4.0.0-<build number>_amd64.deb

  These contain the files of the application graphical user interface. Packages can be installed to 64-bit operating systems based on the type of package manager.

- kess-gui-4.0.0-<build number>.aarch64.rpm, kess-gui-4.0.0-<build number>_arm64.deb

  These contain the files of the application graphical user interface. Packages for the relevant package manager can be installed on 64-bit operating systems for the Arm architecture.

- kess-4.0.0.<build number>.zip

  Contains the files used for remote application installation using Kaspersky Security Center, including license.<language ID> and ksn_license.<language ID> files.

  > Kaspersky Security Center Network Agent is not included in the distribution kit. You can download it on the application download page in the **Kaspersky Security Center** section.

- ksn_license.<language ID>

  This file contains the text of the Statement on Kaspersky Security Network.

- license.<language ID>

  This file contains the text of the End User License Agreement. The End User License Agreement specifies the terms for using the application.

Editing configuration files of the application on your on using means not described in the application documentation or not recommended by Technical Support may cause poor performance and failures of the application and operating system, reduced protection of your device, inaccessible and corrupted data, as well as the sending of additional statistics to KSN getting turned on.

Hardware and software requirements

This section contains the hardware and software requirements for Kaspersky Embedded Systems Security.

Hardware requirements

Kaspersky Embedded Systems Security has the following minimum hardware and software requirements:

Minimum hardware requirements

- CPU with AMD64 architecture, 2 GHz or more, two or more cores, and support for the SSE2 instruction set.
- Swap partition of at least 1 GB
- 2 GB of RAM available to the application for 64-bit operating systems (1 GB of RAM available to the application for 32-bit operating systems).

- 4 GB of free hard disk space for installation of the application and storage of temporary and log files

  Additional disk space is required to store trace files and application memory dump files.

  When scanning archives and compound objects with `ScanArchived=Yes` and/or `ScanSfxArchived=Yes` (by default, these are set to `Yes` for scan tasks and `No` for File Threat Protection), File Threat Protection and scan tasks require additional disk space depending on the type and unpacked size of such objects.

  We recommend making sure there is enough free space in the /tmp directory for scanning archives and compound objects. You can also limit the size (in the archived state) of archives that can be scanned by specifying the `SizeLimit` setting for each task.

- When using the graphical interface, the monitor must be capable of displaying windows 1000 pixels wide and 600 pixels tall (if screen scaling is applied, these dimensions are also scaled).

Minimum hardware requirements for the Arm architecture

- Armv8.2-A Kunpeng 920 or Armv8-A Baikal-M (BE-M1000) CPU or m-TrusT Terminal platform
- Swap partition of at least 1 GB
- 2 GB of RAM available to the application
- 3 GB of free hard disk space for installation of the application and storage of temporary and log files

  Additional disk space is required to store trace files and application memory dump files.

  When scanning archives and compound objects with `ScanArchived=Yes` and/or `ScanSfxArchived=Yes` (by default, these are set to `Yes` for scan tasks and `No` for File Threat Protection), File Threat Protection and scan tasks require additional disk space depending on the type and unpacked size of such objects.

  We recommend making sure there is enough free space in the /tmp directory for scanning archives and compound objects. You can also limit the size (in the archived state) of archives that can be scanned by specifying the `SizeLimit` setting for each task.

- When using the graphical interface, the monitor must be capable of displaying windows 1000 pixels wide and 600 pixels tall (if screen scaling is applied, these dimensions are also scaled).

Software requirements

To install Kaspersky Embedded Systems Security, one of the following operating systems must be installed on the device:

- Supported 32-bit operating systems:
  - Debian GNU/Linux 12.7, 12.8, 12.9, 12.10, 12.11, 12.12.
- Supported 64-bit operating systems:
  - AlmaLinux OS 9.4, 9.5, 9.6.
  - Astra Linux Special Edition RUSB.10015-01 (operational update 1.7).
  - Astra Linux Special Edition RUSB.10015-01 (operational update 1.8).

- Astra Linux Special Edition RUSB.10015-03 (operational update 7.6).
- Astra Linux Special Edition RUSB.10015-17 (operational update 1.7.3)
- Astra Linux Special Edition RUSB.10015-37 (operational update 7.7).

  The application is supported on Astra Linux operating systems in mandatory access control mode if the use of the system namespace mechanism is enabled.

- CentOS Stream 9.
- CentOS Stream 10.
- Debian GNU/Linux 12.7, 12.8, 12.9, 12.10, 12.11, 12.12.
- Debian GNU/Linux 13.0, 13.1, 13.2.
- EulerOS 2.0 SP10.
- Oracle® Linux 9.4, 9.5, 9.6.
- Red Hat® Enterprise Linux® 9.4, 9.5, 9.6.
- Red Hat Enterprise Linux 10.0, 10.1.
- Rocky Linux™ 9.4, 9.5, 9.6.
- SUSE® Linux Enterprise Server 15.4, 15.5, 15.6, 15.7.
- Ubuntu® 22.04 LTS.
- Ubuntu 24.04 LTS.
- ALT Workstation 11.
- ALT Server 11
- ALT SP Workstation release 10.
- ALT SP Server release 10.
- RED OS® 7.3.
- RED OS 8.0.
- ROSA "Cobalt" 7.9 Workstation.
- ROSA "Cobalt" 7.9 Server.
- ROSA "Chrome" 12 Workstation.
- ROSA "Chrome" 12 Server.

- 64-bit operating systems for the Arm architecture:
  - Astra Linux Special Edition RUSB.10152-02 (operational update 4.7).
  - CentOS Stream 9.
  - Debian GNU/Linux 12.13.
  - EulerOS 2.0 SP10.
  - SUSE Linux Enterprise Server 15.4, 15.5, 15.6, 15.7.
  - Ubuntu 22.04 LTS.
  - ALT SP Workstation release 10.
  - ALT SP Server release 10.
  - RED OS 8.0.

Due to technical limitations of fanotify, the application does not support the following file systems: autofs, binfmt_misc, cgroup, configfs, debugfs, devpts, devtmpfs, fuse, fuse.gvfsd-fuse, gfs2, gvfs, hugetlbfs, mqueue, nfsd, proc, parsecfs, pipefs, pstore, usbfs, rpc_pipefs, securityfs, selinuxfs, sysfs, tracefs.

Special considerations involved with using the application in different operating systems

The application is supported on Astra Linux operating systems in mandatory access control mode if the use of the system namespace mechanism is enabled.

The operating system needs additional configuration to support Kaspersky Embedded Systems Security in the following cases:

- Astra Linux Special Edition in closed software environment mode.
- ALT SP in closed software environment mode.
- When using the SELinux access control system on operating systems based on Red Hat Enterprise Linux.

Kaspersky Embedded Systems Security is not supported in the following cases:

- On operating systems based on Red Hat Enterprise Linux, with the IMA subsystem.
- On operating systems based on Red Hat Enterprise Linux, in FIPS mode.
- On Ubuntu operating systems and operating systems based on the SUSE Linux Enterprise Server code base, with the AppArmor mandatory access control system.
- In the Grsecurity security system (on all operating systems).
- When the seccomp mechanism is used (on all operating systems).

- When WSL1 and WSL2 technologies are used (on all operating systems).
- When certain file system mount options are used (for example, noexec, nosuid).
- When Kaspersky Embedded Systems Security is installed inside a container.

In operating systems with auditd 3.0.7 and Linux kernel 4.18, using Behavior Detection may result in the freezing of the application and ssh connections.

Supported versions of Kaspersky Security Center

You can manage Kaspersky Embedded Systems Security using the following versions of Kaspersky Security Center:

- Kaspersky Security Center Windows 14, 14.2, 15.1 You can manage the Kaspersky Embedded Systems Security application in the Administration Console using the administration MMC plug-in and in the Kaspersky Security Center Web Console using the administration web plug-in.
- Kaspersky Security Center Linux 15.1, 15.2, 15.3, 15.4, 16.0, 16.1, 16.2 The administration web plug-in can be used to administer Kaspersky Embedded Systems Security through Kaspersky Security Center Web Console.

Kaspersky Security Center Network Agent is required to manage Kaspersky Embedded Systems Security through Kaspersky Security Center.

Kaspersky Security Center Network Agent is not included in the Kaspersky Embedded Systems Security distribution kit. You can download it on the application download page in the **Kaspersky Security Center** section.

What's new

Kaspersky Embedded Systems Security 4.0 for Linux now has the following new features and improvements:

- The application can now run on Arm devices.
- Introducing a new functional component, BadUSB Attack Prevention. This component prevents the connection of infected USB devices that imitate a keyboard to a client device.
- Integration of Kaspersky Embedded Systems Security with Kaspersky Managed Detection and Response will become available soon. Kaspersky Managed Detection and Response is a solution that automatically detects and analyzes security incidents in your infrastructure. It uses telemetry data received from ATMs and POS terminals, as well as machine learning.
- The Device Control component has been improved:
  - We added the ability to configure a list of users or user groups where added devices are considered trusted.
  - We added the ability to allow read-only access to storage devices.
  - Now you can grant temporary access to blocked devices upon user request.
  - Application management plug-ins now allow adding trusted devices based on events sent to Kaspersky Security Center. The import of lists of trusted devices from a file has also been improved.
  - We also improved user notification through events, adding new events and clarifying and systematizing event texts in plug-ins and on the command line.
- The Application Control component has been improved.
  - Now you can allow running scripts signed by trusted certificates in allowlist mode.
  - New counter of trusted certificates.
- Now you can select the traffic interception mode: with eBPF technology or the iptables utility with related kernel mechanisms.
- Now you can configure traffic interception exclusions. Now you can use the management plug-ins of the application or command line to specify connections that the application should exclude from traffic interception.
- Now you can select the telemetry source (only eBPF or eBPF plus auditd). For auditd, you can also select an operating mode (exclusive mode or multicast mode).
- The policy management interface in the Web Console has been updated. The structure of sections in the policy has been reworked to help find the functions you need quicker.
- Now you can use the Kaspersky Private Security Network (KPSN) directly without using the KSN proxy service on the Administration Server.

- Now you can enable or disable the use of the KSN proxy server for the communication of Kaspersky Embedded Systems Security with KSN servers. If the use of the KSN proxy server is enabled, Kaspersky Embedded Systems Security communicates with KSN servers using the KSN Proxy service on the Administration Server. If the use of the KSN proxy server is disabled, Kaspersky Embedded Systems Security interacts with the KSN servers directly.

- Now you can create out-of-office policies in the Web Console and Administration Console. Out-of-office policies come into effect when a device leaves the organization's network.

- We added the ability to configure the initial configuration settings of the Kaspersky Embedded Systems Security application in the Kaspersky Security Center Administration Console when creating the installation package or in the properties of the installation package.

- Now you can configure the cancellation of scheduled tasks on a device running on battery power.

- Now you can temporarily exclude database log files from scanning to optimize the File Threat Protection scan. If a database log file is reused by the same process within 10 minutes of the last scan, the application skips the scan of that log file.

- Now you can use the management web plug-in to export and import lists of exclusions in policies and tasks.

- We added the ability to display information about the policy and policy profiles in effect on the device on the command line and in the graphical interface.

- When viewing and exporting the list of mount points in the Web Console or Administration Console, or when viewing this list on the command line, information about the mount point being found in a namespace is now also displayed.

- Now you can manage Backup settings in the properties of a managed device.

Version 3.4

Kaspersky Embedded Systems Security 3.4 for Linux now has the following new features and improvements:

- The Application Control component has been improved. Now you can configure execution prevention for the application other than those signed with trusted signatures.

- More settings of the Inventory task. Now you can select an action with the "Golden image" category when a task finishes.

- New cloud mode of the application where the application uses a lightweight version of the anti-malware databases. This helps reduce the load on device memory.

- The application can be automatically restarted after an update. When updating, the application now automatically restarts to save the administrator the additional step of restarting the application.

- New settings that let you limit CPU usage have been added to the general application settings. The `ScanPriority` setting was also removed from *ODS* and *InventoryScan* scan tasks.

- Now you can disable protection components and scan tasks after the application is installed. Installation with protection components disabled can be convenient, for example, to reproduce a problem in the operation of the application and create a trace file.

- The configuration file of the application now includes settings for defining the directory for storing dump files and the minimum required free disk space after creating dump files.

- We added the ability to view the remote application installation log and manage the tracing process in the Web Console properties of the managed device or in the Administration Console using the remote diagnostics utility.

- Now you can export and import a list of trusted devices for the Device Control component in the application administration plug-ins.

- Now you can export and import per-process exclusions for the Behavior Detection component in the application management plug-ins.

- Now you can configure the application to interact directly with KSN servers if the KSN Proxy service becomes unavailable.

- In the application management plug-ins, in the **Storages** -> **Backup** section, you can send a file to Kaspersky for scanning.

- The graphical user interface implements the ability to inform the user about the operation of application components and tasks in "Notify only" mode, in which, if a threat is detected, application components and tasks do not attempt to disinfect or remove malicious objects, deny access, or block the activity of programs, but instead only inform the user that a threat was detected. Administration plug-ins also now have a notification that "Notify only" mode has been enabled for the File Threat Protection and Device Control components.

- The newly implemented application stability monitoring functionality allows you to track the number of times the application terminates abnormally and notify the administrator about the unstable operation of the application.

- Now you can configure the display of pop-up notifications when managing the graphical user interface of the application. Now you can enable or disable the display of pop-up notifications in the GUI using the Web Console, Administration Console, or command line.

- Improved command line options for task scheduling. Now you can configure a task to be stopped when it reaches the maximum allowed execution time.

- Improved Firewall Management component. Now you can specify multiple IP addresses or IP address ranges for a network packet rule.

- Now you can view statistics for the most frequently scanned files and applications, as well as a list of mount points detected on the protected device. To improve performance, you can now exclude the files most frequently scanned by the File Threat Protection component, the paths to applications most frequently scanned by the Behavior Detection component, and the mount points detected on the device.

- Now you can obtain application performance metrics that can help analyze the impact of Kaspersky Embedded Systems Security on operating system performance.

- The procedure for initial configuration of the application in interactive mode now includes a check for the existence of users in privileged groups. The check for the presence of SELinux in the system has also been improved.

- The user is now better informed thanks to new events, improved event texts, an expanded list of event attributes, and a unification of events in plug-ins and the command line.

- Container protection and scanning is not provided as separate functionality. If you had Container Scan tasks configured in the previous version of the application, these tasks become unavailable and are not displayed after updating the application. Scanning files in containers being started is possible as part of File Threat Protection when namespace scanning is enabled.

- We updated the list of supported operating systems.

Preparing to install Kaspersky Embedded Systems Security

Before starting installation of Kaspersky Embedded Systems Security, you need to perform the following actions:

- Check that your device meets the hardware and software requirements of the application.
- Be sure third-party anti-virus software is not installed on your device.
- Be sure that Kaspersky Endpoint Agent for Linux is not installed on your device. If Kaspersky Endpoint Agent for Linux is installed, during the installation process you will see a message about the need to manually remote it.
- If you plan to manage Kaspersky Embedded Systems Security using Kaspersky Security Center, prepare the installation of Kaspersky Security Center Network Agent on the device (for details, see the Kaspersky Security Center Help).
- If you want to use the SELinux access control system, you need to make sure that the semanage utility is installed on the operating system. If the utility is not installed, install the policycoreutils-python or policycoreutils-python-utils package, depending on the package manager.
- If you want to use the graphical user interface, you need to make sure that the dependencies of the GUI package are satisfied on your device. If the device is in an isolated network segment and does not have access to the repositories of the package manager, we recommend checking the list of dependencies on a reference device (libice6, libsm6, libx11-6, libx11-xcb1, libxcb-icccm4, libxcb-image0, libxcb-keysyms1, libxcb-randr0, libxcb-render-util0, libxcb-render0, libxcb-shape0, libxcb-shm0, libxcb-sync1, libxcb-xfixes0, libxcb-xinerama0, libxcb-xkb1, libxcb1, libxi6, libxml2, zlib1g), and then downloading and distributing the packages to all devices before installing the GUI.
- If you want to use the graphical user interface, you need to make sure that the Zenity utility is installed on the operating system.
- On devices running the SUSE Linux Enterprise Server 15 operating system, the insserv-compat package must be installed.
- By default, Astra Linux operating systems block ptrace (Disable ptrace capability), which may affect the operation of Kaspersky Embedded Systems Security. For Kaspersky Embedded Systems Security to work correctly, unblock ptrace when installing Astra Linux. If Astra Linux is already installed, see the Astra Linux Help Center website for instructions on how to enable/disable this mode (**Configuring protection and blocking mechanisms** in the **Blocking ptrace** section).

  You can disable ptrace tracing by specifying the kernel parameter-value pair `kernel.yama.ptrace_scope=3`. In this case, the following will happen in the application:

  - Creating dump files of the kess process when the application crashes will not be possible.
  - The Behavior Detection component will not be able to receive the environment settings of processes.

- On operating systems with a kernel version higher than 5.7, the Application Control component will not be able to read command line arguments and process environment variables, because reading /proc/pid/syscall and /proc/pid/mem is prohibited. On operating systems with kernel version 5.0, the Application Control component will not function properly.

- The process memory and kernel memory scan will not work in scan tasks, since reading / proc / pid / mem is prohibited.

- For the Firewall Management, Web Threat Protection and Network Threat Protection components to work, the iptables utility needs to be installed on your device.

- For the Kaspersky Embedded Systems Security MMC management plug-in to work, Microsoft® Visual C++® 2015 Redistributable Update 3 RC (see https://www.microsoft.com/en-us/download/details.aspx?id=52685) must be installed on device where Kaspersky Security Center Administration Server is installed.

- For the application to run correctly, make sure that the root account is the owner of the following directories and that only the owner has the right to write to them: /var, /var/opt, /var/opt/kaspersky, /var/log/kaspersky, /opt, /opt/kaspersky, /usr/bin, /usr/lib, /usr/lib64.

- Make sure that file descriptor limits recommended by the operating system vendor are configured in the operating system. To check the limit, run the command `cat /proc/sys/fs/file-max`. When the application is running, the operating system may use significantly more descriptors. In general, we recommend disabling the file limit by specifying `fs.file-max=9223372036854775807` in the /etc/sysctl.conf file. After changing the value of this setting, you must restart the operating system.

Installation and initial configuration of Kaspersky Embedded Systems Security

You need to prepare for installation before installing the Kaspersky Embedded Systems Security.

These scenarios describe the installation and post-installation configuration of Kaspersky Embedded Systems Security, the installation and configuration of Kaspersky Security Center Network Agent and the installation of Kaspersky Embedded Systems Security management plug-ins. The installation scenario depends on the mode in which you plan to use Kaspersky Embedded Systems Security.

The application installation procedure involves the following steps:

1. **Installation and initial configuration of the Network Agent**

   If you plan to manage Kaspersky Embedded Systems Security using Kaspersky Security Center, install and configure Kaspersky Security Center Network Agent on the protected device.

2. **Installing the Kaspersky Embedded Systems Security management plug-in**

   If you plan to manage Kaspersky Embedded Systems Security using Kaspersky Security Center, install the Kaspersky Embedded Systems Security management plug-in. Depending on the console used to manage Kaspersky Security Center, the following management plug-ins are used:

   - The Kaspersky Embedded Systems Security administration web plug-in lets you manage the application using Kaspersky Security Center Cloud Console and Kaspersky Security Center Web Console. The web plug-in is installed on the device that has the Kaspersky Security Center Web Console installed.
   - The Kaspersky Embedded Systems Security administration MMC plug-in lets you manage the application using Kaspersky Security Center Administration Console. The MMC plug-in is installed on the device where Kaspersky Security Center Administration Console is installed.

3. **Installing application packages and graphical user interface**

   Kaspersky Embedded Systems Security is distributed in the DEB and RPM packages. There are separate packages for the application and for the graphical user interface. Install Kaspersky Embedded Systems Security and, if necessary, the graphical user interface from packages in the appropriate format.

   You can perform installation in one of the following ways:

   - Using Kaspersky Security Center.
   - Using the command line.

4. **Kaspersky Embedded Systems Security post-installation configuration**

   The application needs initial configuration to prepare it for operation and enable the protection of the client device.

   If you installed Kaspersky Embedded Systems Security using Kaspersky Security Center, the initial configuration was performed automatically during installation in accordance with the parameters specified in the installation package. After completing the installation, go through the Getting started procedure.

   If you installed Kaspersky Embedded Systems Security using the command line, run the initial configuration script or perform the initial configuration in automatic mode after installation is completed.

If initial configuration of the application has not been completed on a device, you cannot use or update the application on that device.

The installation and initial configuration of Kaspersky Security Center Network Agent

Network Agent facilitates the client device's connection with the Kaspersky Security Center Administration Server. It must be installed on every client device that will be connected to Kaspersky Security Center, the centralized remote management system.

Before you begin installing the Network Agent on Linux devices, you must perform some preparatory steps. For instructions on how to prepare devices for installing Network Agent, please refer to the Kaspersky Security Center Help. The procedure depends on the operating system.

You can perform the installation and initial configuration of Network Agent:

- Remotely from the administrator's workstation using the Kaspersky Security Center Web Console or the Administration Console. The Network Agent installation package is used for remote installation.
- Using the command line:
  - In silent mode with an answer file. An answer file is a text file that contains a custom set of settings for the installation and initial configuration of Network Agent. For a description of installation options and initial configuration of Network Agent, please refer to the Kaspersky Security Center Help (the "Installing Network Agent for Linux in silent mode (with an answer file)" section).
  - Interactively from an RPM or DEB package depending on your package manager. In this case, after installation, you need to perform the initial configuration of the Network Agent using a script.

For details on how to install Network Agent, refer to the Kaspersky Security Center Help system.

Installing the Kaspersky Embedded Systems Security management plug-ins

The following Kaspersky Embedded Systems Security management plug-ins are used to manage Kaspersky Embedded Systems Security using Kaspersky Security Center:

- The Kaspersky Embedded Systems Security administration web plug-in lets you manage the application using Kaspersky Security Center Cloud Console and Kaspersky Security Center Web Console.
- The Kaspersky Embedded Systems Security administration MMC plug-in lets you manage the application using Kaspersky Security Center Administration Console.

You can install management plug-ins for different versions of Kaspersky Embedded Systems Security simultaneously. This allows you to manage the application by using the policies created with different administration plug-in versions.

If you use the Kaspersky Security Center Administration Console to manage the application, you can also convert policies and tasks created using earlier versions of the management plug-in to newer versions.

Installing the Kaspersky Embedded Systems Security web plug-in

The Kaspersky Endpoint Security management web plug-in must be installed on the device that has the Kaspersky Security Center Web Console installed. The functionality of the web plug-in is available to all administrators who have access to Kaspersky Security Center Web Console in a browser.

You can install the web plug-in as follows:

- Using the Quick Start Wizard for Kaspersky Security Center Web Console.

  Kaspersky Security Center Web Console automatically prompts you to run the Quick Start Wizard when connecting Kaspersky Security Center Web Console to the Administration Server for the first time. You can also run the Initial Configuration Wizard in the Kaspersky Security Center Web Console interface (**Device discovery and deployment** → **Deployment and assignment** → **Quick Start Wizard**). The Quick Start Wizard can also check if the installed web plug-ins are up to date and download the necessary updates. For more information on the Initial Configuration Wizard for Kaspersky Security Center Web Console, please refer to Kaspersky Security Center Help section.

- Manually, using a distribution kit from the list of Kaspersky Web plug-ins or from an external source.

To install the Kaspersky Embedded Systems Security web plug-in manually:

1. In the main window of the Kaspersky Security Center Web Console, select **Settings** → **Web plug-ins**.

   A list of installed web plug-ins opens.

2. Start the installation of the Kaspersky Embedded Systems Security web plug-in by one of the following ways:

   - Installation from the list of Kaspersky web plug-ins:

     a. Click the **Add** button.

        A list of all available Kaspersky Web plug-ins opens. The list is updated automatically after new versions of web plug-ins are released.

     b. Find the **Kaspersky Embedded Systems Security <version number> for Linux** web plug-in in the list and click its name.

     c. In the window that opens with a description of the web plug-in, click the **Install plug-in** button.

     d. Wait for the installation to complete and click **OK** in the information window.

     e. This opens the **Web plug-in settings** window.

        > In Kaspersky Security Center Web Console 14.2, you need to refresh the browser page to open the **Web plug-in settings** window.

     f. Wait for the installation to complete and click **OK** in the information window.

- Installing a web plug-in from a third-party source (archives required for installing web plug-ins are included in the distribution kit or can be downloaded from the Kaspersky website):

  a. Click the **Add from file** button.

  b. In the window that opens, specify the path to the ZIP archive with the distribution kit for the web plug-in and the path to the signed file in TXT format. This file is in the archive with the web plug-in.

  c. Click the **Add** button.

  d. Wait for the installation to complete and click **OK** in the information window.

  e. This opens the **Web plug-in settings** window.

  > In Kaspersky Security Center Web Console 14.2, you need to refresh the browser page to open the **Web plug-in settings** window.

  f. Wait for the installation to complete and click **OK** in the information window.

The new plug-in is displayed in the list of installed web plug-ins (**Settings** → **Web Plug-ins**).

If you select a language that is not included in Kaspersky Embedded Systems Security distribution package in the properties of Kaspersky Security Center Administration Server, the License Agreement and the entire Kaspersky Security Center Web Console interface will be displayed in English.

Installing the Kaspersky Embedded Systems Security MMC plug-in

The Kaspersky Endpoint Security management MMC plug-in must be installed on the same device where the Kaspersky Security Center Administration Console is installed.

Before installing the Kaspersky Embedded Systems Security administration MMC plug-in, make sure that Kaspersky Security Center and Redist C++ 2015 (Microsoft Visual C++ 2015 Redistributable) are installed.

To install the MMC plug-in:

on the device where the Kaspersky Security Center Administration Console is installed, run the executable file klcfginst.msi.

The file is included in the Kaspersky Embedded Systems Security distribution kit.

After installation, the MMC management plug-in is displayed in the list of installed MMC management plug-ins in the properties of the Kaspersky Security Center Administration Server.

To view the list of installed management MMC plug-ins:

1. In the Kaspersky Security Center Administration Console tree, select the **Administration Server <server name>** node and open the Administration Server properties window in one of the following ways:

   - using the **Properties** item in the **Administration Server <server name>** node context menu;
   - by clicking the **Administration Server properties** link located in the workspace of the **Administration Server <server name>** node in the **Administration Server** section.

2. In the list on the left, in the **Advanced** section, select the **Information about the installed application management plug-ins** section.

   In the right part of the window, the list of installed management plug-ins displays the administration MMC plug-in for Kaspersky Embedded Systems Security: **Kaspersky Embedded Systems Security <version number> for Linux**.

Installing and initially configuring the application using Kaspersky Security Center

You can install Kaspersky Embedded Systems Security on a client device remotely from the administrator's workstation using the Kaspersky Security Center Web Console or the Administration Console.

Installation using Kaspersky Security Center involves the following steps:

1. Creating an installation package.

   For the remote installation, Kaspersky Embedded Systems Security installation package is used. The Kaspersky Embedded Systems Security installation package is the same for all supported operating systems and processor architecture types. You can create the installation package using the Kaspersky Security Center Web Console or the Administration Console.

   You can specify the initial configuration settings using the autoinstall.ini configuration file included in the installation package, or in the properties of the installation package.

   You can add the following to the installation package that you are creating:

   - License key for automatic activation of the application during installation
   - Pre-downloaded application databases to avoid having to update the databases after installation

   You can also activate the application and update the databases as part of the getting started procedure

2. Deploying the Kaspersky Endpoint Security application on devices in the corporate network.

   Kaspersky Security Center Web Console supports the following main deployment methods:

   - Installing the application using the Protection Deployment Wizard.
   - Installing the application using the remote installation task.

   The Kaspersky Security Center Administration Console supports the following main deployment methods:

   - Installing the application using the Remote Installation Wizard.
   - Installing the application using the remote installation task.

For a description of the deployment procedures, see the Kaspersky Security Center Help.

If necessary, you can view the application remote installation log by using remote diagnostics of the Kaspersky Security Center client device.

3. Getting started.

   Before using the application, you need to complete the initial configuration of the application and prepare the application for operation.

   If initial configuration of the application has not been completed on a device, you cannot use or update the application on that device.

To use Kaspersky Security Center to manage Kaspersky Embedded Systems Security installed on client devices, you need to put these devices in administration groups. Before starting Kaspersky Embedded Systems Security installation, you can create Kaspersky Security Center administration groups to which you want to move the devices with the application installed, and configure the rules to automatically move the devices to these administration groups. If rules for moving devices to the administration groups are not configured, Kaspersky Security Center moves all the devices that have the Administration Agent installed and are connected to Administration Server to the **Unassigned devices** list. In this case, you need to manually move computers to the administration groups (refer to the Kaspersky Security Center Help for details).

Creating an installation package in the Web Console

In Kaspersky Security Center Web Console, you can create an installation package in one of the following ways:

- From an archive file that you have prepared previously.
- From a distribution kit hosted on Kaspersky servers.

To prepare an archive for creating an installation package:

1. Download the kess.zip archive from the application download page. It is located in the **Kaspersky Embedded Systems Security for Linux** -> **Additional distribution** → **Files for Product remote installation** section.
2. Unpack the kess.zip archive to a folder accessible to Kaspersky Security Center Administration Server. Place the distribution files, that correspond to the type of operating system where you want to install the application and the type of its package manager, to the same folder:
   - To install Kaspersky Embedded Systems Security:
     - kess-4.0.0-<build number>.i386.rpm (for 32-bit operating systems with rpm)
     - kess_4.0.0-<build number>_i386.deb (for 32-bit operating systems with dpkg)
     - kess_4.0.0-<build number>.x86_64.rpm (for 64-bit operating systems with rpm)
     - kess_4.0.0-<build number>_amd64.deb (for 64-bit operating systems with dpkg)
     - kess-4.0.0-<build number>.aarch64.rpm (for 64-bit operating systems with rpm for the Arm architecture)
     - kess_4.0.0-<build number>_arm64.deb (for 64-bit operating systems with dpkg for the Arm architecture)

- To install the graphical user interface of the application:
  - kess-gui-4.0.0-<build number>.i386.rpm (for 32-bit operating systems with rpm)
  - kess-gui-4.0.0-<build number>_i386.deb (for 32-bit operating systems with dpkg)
  - kess-gui-4.0.0-<build number>.x86_64.rpm (for 64-bit operating systems with rpm)
  - kess-gui-4.0.0-<build number>_amd64.deb (for 64-bit operating systems with dpkg)
  - kess-gui-4.0.0-<build number>.aarch64.rpm (for 64-bit operating systems with rpm for the Arm architecture)
  - kess-gui-4.0.0-<build number>_arm64.deb (for 64-bit operating systems with dpkg for the Arm architecture)

  If you do not want to install the graphical user interface, do not add these files to the folder; this will make the installation package smaller.

If you do not plan to use the graphical interface, disable it by editing the appropriate setting (`USE_GUI=No`) in the properties of the created installation package or in the autoinstall.ini configuration file. Otherwise, the installation will fail.

If you want to use the created installation package with different operating systems or package managers, place the files for all the types of operating systems and package managers that you need in the directory.

3. If you want to use offline application databases downloaded in advance:
   a. Place prepared archives with databases for all your operating system types into the folder.
   b. In initial configuration settings, disable the database update task after installing the application. You can configure the relevant setting in the properties of the created installation package or in the autoinstall.ini configuration file (`UPDATE_EXECUTE=no`). The autoinstall.ini file is located in the directory where you extracted the kess.zip archive.
4. If you want to perform the initial configuration of the application using a configuration file, open the autoinstall.ini configuration file and edit it as necessary.

   You can also perform the initial configuration of the application later in the properties of the created installation package on the **Settings** tab.
5. Place all prepared files in an archive in ZIP, CAB, TAR, or TAR.GZ format with any name.

To create an installation package for Kaspersky Embedded Systems Security in Kaspersky Security Center Web Console:

1. In the main Web Console window, select one of the following sections:
   - **Device discovery and deployment** → **Deployment and assignment** → **Installation packages**.
   - **Operations** → **Repositories** → **Installation packages**.

   A list of installation packages available on the Administration Server opens.

2. Click the **Add** button.

   The wizard for creating an installation package will start.

3. Select a method that you want to use to create the installation package:

   - **Create an installation package from a file**. The installation package will be created from an archive that you have prepared in advance.
   - **Create the installation package for a Kaspersky application**. The installation package will be created from a distribution package located on Kaspersky servers.

   Kaspersky Security Center Cloud Console does not allow creation of installation packages from a file.

4. Depending on the selected package creation method:

   - Specify the package name, click the **Browse** button, and specify the path to the archive that you have prepared for creating the installation package.
   - Select Kaspersky Embedded Systems Security distribution package. In the window on the right, read the information about the distribution package and click the **Download and create installation package** button. The installation package creation process starts.

5. Read the End User License Agreement concluded between you and Kaspersky, and the Privacy Policy describing the handling and transmission of data. To continue creating the installation package, you must confirm that you have read and accept the full terms of the End User License Agreement and the Privacy Policy.

6. Finish the wizard.

   The installation package will be created and added to the list of installation packages. Using the installation package, you can install the application on devices in the corporate network or update the application version.

7. If necessary, manage the initial configuration settings. To do this, open the properties of the installation package and go to **Settings** tab.

   Kaspersky Security Center Web Console versions earlier than 14.2 do not support providing initial configuration settings in the properties of the Kaspersky Embedded Systems Security installation package. Use the autoinstall.ini configuration file to configure settings.

Initial configuration settings in the Web Console

In Kaspersky Security Center Web Console, you can specify the initial configuration settings for Kaspersky Embedded Systems Security in the installation package properties on the **Settings** tab. The created installation packages are available in the main window of the Web Console in one of the following sections:

- **Device discovery and deployment** → **Deployment and assignment** → **Installation packages**.
- **Operations** → **Repositories** → **Installation packages**.

You can also specify initial configuration settings of the application in the configuration file that is included in the installation package.

Setting	Description
Language of events in KSC	Select the localization language for application events sent to Kaspersky Security Center: • **System** (default) to use the operating system locale. If the application cannot detect the locale of the operating system or this locale is not supported, English is used to localize events. • One of the languages supported by Kaspersky Embedded Systems Security. The application uses the UTF-8 encoding. The locale of the graphical interface and the application command line is taken from the `LANG` environment variable. If the application does not support this localization, it defaults to the English localization.
Activate the application	Select this check box if you want to activate the application during installation. In the displayed field, enter the activation code. This check box is cleared by default. You can also activate the application after installation.
Update source	Select the update source for databases and application modules: • **Kaspersky update servers** (default). • **Kaspersky Security Center**. • **Other source in the local or global network**. If you select this option, enter the address of the update source in the field that opens.
Run database update task after installation	If this check box is selected, after the application is installed, the databases and application modules update task starts. This check box is selected by default.
Specify the proxy server settings	Select this check box if you use a proxy server for internet access. In the displayed field, enter the proxy server address in one of the following formats: • `[<connection protocol>://]<IP address of the proxy server>:<port>`, if no authentication is required to connect to the proxy server; • `[<connection protocol>://][<user name>:[<password>]@]<IP address of the proxy server>:<port>`, if authentication is required when connecting to the proxy server. Connecting to a proxy server over HTTPS is not supported.
Install kernel source	If this check box is selected, starting File Threat Protection automatically starts the compilation of the kernel module on operating systems that do not support fanotify. This check box is selected by default.
Use the graphical user interface	Select this check box if you plan to install and use the application graphical interface. Files for installing the graphical interface must be added to the installation package. This check box is cleared by default.
Specify a user with the admin role	Select this check box to specify the user to which you want to assign the admin role. The Administrator role allows managing application settings and task settings in the graphical interface of the application and on the command line without using the `sudo` command. In the displayed field, enter the user name.
Configure SELinux automatically	If this check box is selected SELinux is automatically configured to work with Kaspersky Embedded Systems Security. This check box is selected by default.
Remove users from privileged groups	Select this check box to automatically remove users from the 'kessadmin' and 'kessaudit' privileged groups before installing the application. This check box is cleared by default. If the check box is selected and the 'nogroup' group does not exist, the installation fails and you are prompted to manually remove users from privileged groups.

Setting	Description
Disable protection components and scan tasks for the first application startup after installation	Select this check box if, after completing the installation process, you want to run the application with protection components and scan tasks disabled. Installation with protection components disabled can be convenient, for example, to reproduce a problem in the operation of the application and create a trace file. If you enable the necessary components and tasks, the enabled components and tasks will continue to work after the application is restarted.
Enable tracing when the application is started for the first time after installation	Select the check box if you want to create a trace file when the application is started for the first time after installation. To create a trace, a kess_startup.log file is automatically created in the /var/log/kaspersky/kess/ directory. The size limit for regular application trace files does not apply to this file. Tracing stops after five minutes or after the File Threat Protection task is started.

Creating an installation package in the Administration Console

Before creating an installation package for Kaspersky Embedded Systems Security, you need to prepare the files to be included in the package.

To prepare files for creating an installation package:

1. Download the kess.zip archive from the application download page. It is located in the **Kaspersky Embedded Systems Security for Linux** -> **Additional distribution** → **Files for Product remote installation** section.

2. Unpack the kess.zip archive to a folder accessible to Kaspersky Security Center Administration Server. Place the distribution files, that correspond to the type of operating system where you want to install the application and the type of its package manager, to the same folder:

   - To install Kaspersky Embedded Systems Security:
     - kess-4.0.0-<build number>.i386.rpm (for 32-bit operating systems with rpm)
     - kess_4.0.0-<build number>_i386.deb (for 32-bit operating systems with dpkg)
     - kess_4.0.0-<build number>.x86_64.rpm (for 64-bit operating systems with rpm)
     - kess_4.0.0-<build number>_amd64.deb (for 64-bit operating systems with dpkg)
     - kess-4.0.0-<build number>.aarch64.rpm (for 64-bit operating systems with rpm for the Arm architecture)
     - kess_4.0.0-<build number>_arm64.deb (for 64-bit operating systems with dpkg for the Arm architecture)
   - To install the graphical user interface of the application:
     - kess-gui-4.0.0-<build number>.i386.rpm (for 32-bit operating systems with rpm)
     - kess-gui-4.0.0-<build number>_i386.deb (for 32-bit operating systems with dpkg)
     - kess-gui-4.0.0-<build number>.x86_64.rpm (for 64-bit operating systems with rpm)
     - kess-gui-4.0.0-<build number>_amd64.deb (for 64-bit operating systems with dpkg)

- kess-gui-4.0.0-<build number>.aarch64.rpm (for 64-bit operating systems with rpm for the Arm architecture)
- kess-gui-4.0.0-<build number>_arm64.deb (for 64-bit operating systems with dpkg for the Arm architecture)

If you do not want to install the graphical user interface, do not add these files to the folder; this will make the installation package smaller.

If you do not plan to install the graphical interface, you need to opt out by setting `USE_GUI=No` in the autoinstall.ini configuration file. Otherwise, the installation will fail.

If you want to use the created installation package with different operating systems or package managers, place the files for all the types of operating systems and package managers that you need in the directory.

3. If you want to use offline application databases downloaded in advance:

   a. Place prepared archives with databases for all your operating system types into the folder.

   b. In initial configuration settings, disable the database update task after installing the application. To do this, open the autoinstall.ini configuration file and set `UPDATE_EXECUTE=no`. The autoinstall.ini file is located in the directory where you extracted the kess.zip archive.

   If you want to prepare the initial configuration settings of the application, open the autoinstall.ini configuration file and edit it as necessary.

To create an installation package for Kaspersky Embedded Systems Security in the Administration Console of Kaspersky Security Center:

1. In the console tree, select **Additional** → **Remote installation** → **Installation packages**.
2. Click the **Create installation package** button.

   The wizard for creating an installation package will start.
3. Click the **Create an installation package for a Kaspersky application** button.
4. Enter the name of the new installation package and proceed to the next step of the wizard.
5. Select Kaspersky Embedded Systems Security distribution package. To do this, open a standard browsing window using the **Browse** button and specify the path to the kess.kud file. The file is located in the directory where you extracted the kess.zip archive.

   The name and version of the application are displayed in the window.

   Proceed to the next step of the wizard.
6. Read the End User License Agreement concluded between you and Kaspersky, and the Privacy Policy describing the handling and transmission of data.

   To continue creating the installation package, you must confirm that you have read and accept the full terms of the End User License Agreement and the Privacy Policy. To confirm, in the window that opens, select both check boxes.

   Proceed to the next step of the wizard.

7. If necessary, manage the initial configuration settings.

   You can also perform the initial configuration of the application later in the properties of the created installation package in the **Settings** section.

8. The wizard downloads the files required to install the application to Kaspersky Security Center Administration Server. Wait for the download to finish.

9. Finish the wizard.

The created installation package is located in the tree of the Administration Console of Kaspersky Security Center in the **Additional** → **Remote installation** → **Installation packages** folder. You can use the same installation package many times.

Initial configuration settings in the Administration Console

In the Kaspersky Security Center Administration Console, you can specify the initial configuration settings for Kaspersky Embedded Systems Security when creating an installation package or in the installation package properties, in the **Settings** section. The created installation packages are available in the tree of the Administration Console in the **Additional** → **Remote installation** → **Installation packages** folder.

You can also specify initial configuration settings of the application in the configuration file that is included in the installation package.

Initial configuration settings

Setting	Description
Language of events in KSC	Select the localization language for application events sent to Kaspersky Security Center: • **System** (default) to use the operating system locale. If the application cannot detect the locale of the operating system or this locale is not supported, English is used by default. • An explicitly specified language. The application uses the UTF-8 encoding. The locale of the graphical interface and the application command line is taken from the `LANG` environment variable. If the application does not support this localization, it defaults to the English localization.
Activation code	If you want to activate the application during installation, enter the activation code. You can also activate the application after installation.
Select update source	Clicking the **Configure** button opens a window in which you can select the databases and application modules update source: • **Kaspersky update servers** (default). • **Kaspersky Security Center**. • **Other source in the local or global network**. If you select this option, enter the address of the update source in the **Address** field.
Run database update task after installation	If this check box is selected, after the application is installed, the databases and application modules update task starts. This check box is selected by default.

Setting	Description
Proxy server address	If you use a proxy server to access the internet, enter the address of your proxy server in one of the following formats: • `[<connection protocol>://]<IP address of the proxy server>:<port>`, if no authentication is required to connect to the proxy server; • `[<connection protocol>://][<user name>:[<password>]@]<IP address of the proxy server>:<port>`, if authentication is required when connecting to the proxy server. Connecting to a proxy server over HTTPS is not supported.
Install kernel source	If this check box is selected, starting File Threat Protection automatically starts the compilation of the kernel module on operating systems that do not support fanotify. This check box is selected by default.
Use the graphical user interface	Select this check box if you plan to install and use the application graphical interface. Files for installing the graphical interface must be added to the installation package. This check box is cleared by default.
User with admin role	Enter the user to which you want to assign the admin role. The Administrator role allows managing application settings and task settings in the graphical interface of the application and on the command line without using the `sudo` command.
Configure SELinux automatically	If this check box is selected SELinux is automatically configured to work with Kaspersky Embedded Systems Security. This check box is selected by default.
Remove users from privileged groups	Select this check box to automatically remove users from the 'kessadmin' and 'kessaudit' privileged groups before installing the application. This check box is cleared by default. If the check box is selected and the 'nogroup' group does not exist, the installation fails and you are prompted to manually remove users from privileged groups.
Disable protection components and scan tasks for the first application startup after installation	Select this check box if, after completing the installation process, you want to run the application with protection components and scan tasks disabled. Installation with protection components disabled can be convenient, for example, to reproduce a problem in the operation of the application and create a trace file. If you enable the necessary components and tasks, the enabled components and tasks will continue to work after the application is restarted.
Enable tracing when the application is started for the first time after installation	Select the check box if you want to create a trace file when the application is started for the first time after installation. To create a trace, a kess_startup.log file is automatically created in the /var/log/kaspersky/kess/ directory. The size limit for regular application trace files does not apply to this file. Tracing stops after five minutes or after the File Threat Protection task is started.

Preparing an archive with application databases in order to create an installation package with integrated databases

You can create an installation package for remote installation and include pre-downloaded application databases in it. This may be useful, for example, if you are installing the application on a device with the Astra Linux Special Edition operating system. If you are using an installation package with integrated databases, the application is installed with the databases already functional; in this case, you do not need to update the databases immediately after installation.

To create an archive with databases for installing the application:

1. Install and perform the initial configuration of Kaspersky Embedded Systems Security on the device using the command line or using Kaspersky Security Center.

2. Update the application databases. You can update the databases during the initial configuration of the application or after installation by running a task of an Update type in the command line or an *Update* task in the Kaspersky Security Center Administration Console or the Kaspersky Security Center Web Console.

3. Copy the contents of the /var/opt/kaspersky/kess/private/updates/ directory to one of the following subdirectories, depending on the architecture of the operating system for which you are creating the installation package with integrated databases: /i386/, /x86_64/, or /arm64/.

4. Place the directories with the databases into a kess-bases.tgz archive, preserving the structure of nested directories. You can place only one subdirectory with databases for the required architecture of the operating system in the archive, or if you plan to create an installation package for installation on several operating systems with different architectures, you can place all the subdirectories with databases (/i386/, /x86_64/, or /arm64/) into a single archive for different architectures.

5. You can use the created archive with application databases when creating an installation package in the Kaspersky Security Center Administration Console or Kaspersky Security Center Web Console.

Autoinstall.ini configuration file settings

In the autoinstall.ini configuration file, you can specify the settings shown in the table below. The set of applicable settings depends on the application usage mode.

Autoinstall.ini configuration file settings

Setting	Description	Values
EULA_AGREED	Required setting. Acceptance of the terms of the End User License Agreement.	yes (default) – accept the terms and conditions of the End User License Agreement. no – do not accept the terms and conditions of the End User License Agreement. The application installation will be terminated.
PRIVACY_POLICY_AGREED	Required setting. Acceptance of the terms of the Privacy Policy.	yes (default) – accept the terms and conditions of the Privacy Policy. no – do not accept the terms and conditions of the Privacy Policy. The application installation will be terminated.
USE_KSN	Required setting. Enabling Kaspersky Security Network usage: To enable the use of KSN, the terms of the Kaspersky Security Network Statement must be accepted.	yes – accept the terms of the Kaspersky Security Network Statement and enable the use of KSN. no – (default value) do not accept the terms and conditions of the Kaspersky Security Network Statement. Opting out of the KSN does not abort the application installation process. If KSN is enabled, the cloud mode is automatically enabled, in which Kaspersky Embedded Systems Security uses the lightweight version of malware databases.
GROUP_CLEAN	Required setting. Automatically removing users from the kessadmin and kessadmin privileged groups before installing the application.	yes - Remove users from the privileged groups. If the value is yes and there is no nogroup group, the installation will fail and you will be prompted to manually remove users from privileged groups. no - Do not remove users from the privileged groups.

Setting	Description	Values
LOCALE	Optional setting. The locale used for the application events sent to Kaspersky Security Center. The locale of the graphical interface and the application command line is taken from the LANG environment variable. If the application does not support this localization, it defaults to the English localization.	The locale in the format specified by RFC 3066. If the `LOCALE` setting is not specified, the operating system locale is used. If the application fails to determine the operating system localization language or the operating system localization is not supported, the default value will be used – `en_US.utf8`.
INSTALL_LICENSE	Activation code for adding a license key to the application.	`<activation code>` if you want to activate the application during installation. `None` if you do not want to activate the application during installation, for example, if you are installing the application in an isolated network segment without internet access.
UPDATER_SOURCE	Update source for databases and application modules.	`SCServer` – use the Kaspersky Security Center Administration Server as the update source. `KLServers` – use Kaspersky servers as the update source. This value is used by default. `< update source address >`.
PROXY_SERVER	Address of the proxy server used to connect to the Internet.	Address of the proxy server in one of the following formats: • `[<connection protocol>://]<IP address of the proxy server>:<port>`, if no authentication is required to connect to the proxy server; • `[<connection protocol>://][<user name>:[<password>]@]<IP address of the proxy server>:<port>`, if authentication is required when connecting to the proxy server. Connecting to a proxy server over HTTPS is not supported.
UPDATE_EXECUTE	Start the application database update task during the initial configuration.	`yes` (default value) – start the update task. `no` – do not start update task.
KERNEL_SRCS_INSTALL	Automatically start the compilation of the kernel module when the File Threat Protection task is started (for operating systems that do not support the fanotify technology).	`yes` (default) – compile the kernel module when the File Threat Protection task starts. `no` – do not compile the kernel module.
USE_GUI	Installing and using the graphical interface of the application.	`yes` – install and use the application's graphical interface (the files for installing the graphical interface are included in the installation package). `no` (default) – do not use the graphical interface.
ADMIN_USER	A user assigned the administrator role (admin). The Administrator role allows managing application settings and task settings in the graphical interface of the application and on the command line without using the `sudo` command.	

Setting	Description	Values
CONFIGURE_SELINUX	Automatic configuration of SELinux for working with Kaspersky Embedded Systems Security.	`yes` (default) – automatically configure SELinux to work with Kaspersky Embedded Systems Security. `no` – do not automatically configure SELinux to work with Kaspersky Embedded Systems Security.
DISABLE_PROTECTION	Disabling the functional components of the application after installation. An installation with components disabled can be convenient, for example, if you need to reproduce a problem with the application and create a trace file. If you enable the necessary components after installing the application with `DISABLE_PROTECTION=yes`, the enabled components will continue to work after the application is restarted.	`yes` - Disable protection components and scan tasks when the application is started after installation. `no` - Do not disable protection components and scan tasks when the application is started after installation.
ENABLE_TRACES_ON_FIRST_STARTUP	Create a trace file when the application is started for the first time after installation. To create a trace, a kess_startup.log file is automatically created in the /var/log/kaspersky/kess/ directory. The size limit for regular application trace files does not apply to this file. Tracing stops after five minutes or after the File Threat Protection task is started.	`yes` – create a trace file when the application is started for the first time after installation. `no` – do not create a trace file when the application is started for the first time.

If you want to change the settings in the autoinstall.ini configuration file, specify the values of settings in the following format: <setting_name>=<setting_value> (the application does not process spaces between the name of a setting and its value).

Getting started using Kaspersky Security Center

After deploying Kaspersky Embedded Systems Security through Kaspersky Security Center, you must prepare the application for operation. To do so:

1. Activate the application if activation was not performed using the key added to the installation package of the application.

   You can create and execute an activation task using the Administration Console or Kaspersky Security Center Web Console, as well as distribute the license key from the Kaspersky Security Center key storage to the devices.

2. Update the databases and application modules if you did not add pre-downloaded application databases to the installation package of the application. You can use the *Update* task, which is created automatically by the initial configuration wizard of Kaspersky Security Center after installing the administration MMC plug-in or the Kaspersky Embedded Systems Security administration web plug-in.

   Kaspersky Embedded Systems Security protects the device only after the application databases are updated.

3. Configure a policy for centralized management of the application using the Kaspersky Security Center Administration Console (Managing policies in the Administration Console, Creating a policy using the Administration Console) or Web Console. You can use a policy that is created automatically by the initial configuration wizard of Kaspersky Security Center after installing the administration MMC plug-in or the Kaspersky Embedded Systems Security administration web plug-in.

   You can also configure the application management tasks using the Administration Console or the Web Console.

Installing and initially configuring the application using the command line

You can install the Kaspersky Embedded Systems Security application on a client device using the command line.

Installation using the command line involves the following steps:

1. Installing the application and the graphical interface of the application. You can choose one of the following installation options:

   - Install the application with the graphical interface.
   - Install the application without the graphical interface.
   - Install the graphical interface on the device where the application is installed.

     It is not possible to install the graphical interface on a device on which the application is not installed.

   If the version of the apt package manager is lower than 1.1.X, you need to use the dpkg package manager (depending on the operating system) for installation.

2. Initial configuration of the application

   The application needs initial configuration to prepare it for operation and enable the protection of the client device.

   If initial configuration of the application has not been completed on a device, you cannot use or update the application on that device.

   Initial configuration of the application is performed by running the special initial configuration script from the distribution kit of Kaspersky Embedded Systems Security. You can perform the initial configuration of the application in interactive mode or in automatic mode.

Installing the application using the command line

Installing the application without the graphical interface.

To install Kaspersky Embedded Systems Security from an RPM package on a 32-bit operating system, execute the following command:

```
# rpm -i kess-4.0.0-<build number>.i386.rpm
```

To install Kaspersky Embedded Systems Security from an RPM package on a 64-bit operating system, execute the following command:

```
# rpm -i kess-4.0.0-<build number>.x86_64.rpm
```

To install Kaspersky Embedded Systems Security from an RPM package on a 64-bit operating system for the Arm architecture, execute the following command:

```
# rpm -i kess-4.0.0-<build number>.aarch64.rpm
```

To install Kaspersky Embedded Systems Security from a DEB package on a 32-bit operating system, execute the following command:

```
# apt-get install ./kess_4.0.0-<build number>_i386.deb
```

To install Kaspersky Embedded Systems Security from a DEB package on a 64-bit operating system, execute the following command:

```
# apt-get install ./kess_4.0.0-<build number>_amd64.deb
```

To install Kaspersky Embedded Systems Security from a DEB package on a 64-bit operating system for the Arm architecture, execute the following command:

```
# apt-get install ./kess_4.0.0-<build number>_arm64.deb
```

Installing the graphical interface of the application

To install the graphical interface from the RPM package to a 32-bit operating system, execute the following command:

```
# rpm -i kess-gui-4.0.0-<build number>.i386.rpm
```

To install the graphical interface from the RPM package to a 64-bit operating system, execute the following command:

```
# rpm -i kess-gui-4.0.0-<build number>.x86_64.rpm
```

To install the graphical interface from an RPM package on a 64-bit operating system for the Arm architecture, execute the following command:

```
# rpm -i kess-gui-4.0.0-<build number>.aarch64.rpm
```

To install the graphical interface from the DEB package to a 32-bit operating system, execute the following command:

```
# apt-get install ./kess-gui_4.0.0-<build number>_i386.deb
```

To install the graphical interface from the DEB package to a 64-bit operating system, execute the following command:

```
# apt-get install ./kess-gui_4.0.0-<build number>_amd64.deb
```

To install the graphical interface from a DEB package on a 64-bit operating system for the Arm architecture, execute the following command:

```
# apt-get install ./kess-gui_4.0.0-<build number>_arm64.deb
```

Post-installation configuration of the application in interactive mode

To perform initial configuration of the application in interactive mode, you need to run the Initial Configuration Wizard of the Kaspersky Embedded Systems Security application.

You must run the Initial Configuration Wizard as root.

To run the Initial Configuration Wizard, execute the following command:

```
# /opt/kaspersky/kess/bin/kess-setup.pl
```

The wizard requests the values of Kaspersky Embedded Systems Security settings step-by-step. The wizard finishing and the console being released indicate that the post-installation configuration is completed.

To check the return code, execute the following command:

```
echo $?
```

If the command returns 0, the post-installation configuration of the application is completed successfully.

Kaspersky Embedded Systems Security can protect the device only after the application databases are updated.

Selecting the locale

At this step, the application displays the list of supported locales in RFC 3066 format.

Specify the locale as it is listed here. This locale is used for the following purposes:

- Localization of the texts of the End User License Agreement, Privacy Policy, and Kaspersky Security Network Statement, which are displayed during the installation.
- Localization of application events sent to Kaspersky Security Center.

The locale of the graphical interface and the application command line is taken from the `LANG` environment variable. If the application does not support this localization, it defaults to the English localization.

Viewing the End User License Agreement and the Privacy Policy

At this step, read the End User License Agreement concluded between you and Kaspersky, and the Privacy Policy describing the handling and transmission of data.

Accepting the End User License Agreement

At this step, you must either accept or decline the terms of the End User License Agreement.

After exiting viewing mode, enter one of the following values:

- `yes` (or `y`), if you accept the terms of the End User License Agreement.
- `no` (or `n`), if you do not accept the terms of the End User License Agreement.

If you did not accept the terms and conditions of the End User License Agreement, the Kaspersky Embedded Systems Security setup process is aborted.

Accepting the Privacy Policy

At this step, you must either accept or decline the terms of the Privacy Policy.

After exiting viewing mode, enter one of the following values:

- `yes` (or `y`), if you accept the terms of the Privacy Policy.
- `no` (or `n`), if you do not accept the terms of the Privacy Policy.

If you did not accept the terms and conditions of the Privacy Policy, the Kaspersky Embedded Systems Security setup process is aborted.

Using Kaspersky Security Network

At this step, you must either accept or decline the terms of use of the Kaspersky Security Network Statement. The file ksn_license.<language ID> containing the text of the Kaspersky Security Network Statement is located in the directory /opt/kaspersky/kess/doc/.

Enter one of the following values:

- `yes` (or `y`), if you accept the terms of the Kaspersky Security Network Statement. This enables the extended KSN mode.
- `no` (or `n`), if you do not accept the terms of the Kaspersky Security Network Statement.

Refusal to participate in Kaspersky Security Network does not interrupt the initial configuration of Kaspersky Embedded Systems Security. You can enable, disable, or change the Kaspersky Security Network mode at any time.

If Kaspersky Security Network is enabled, the cloud mode is automatically enabled, in which Kaspersky Embedded Systems Security uses the lightweight version of malware databases.

Removing users from privileged groups

This step is displayed only if users are detected in the kessadmin group and/or in the kessaudit group.

At this step, specify whether or not to remove users from the kessadmin and kessaudit privileged groups. Users included in the kessadmin and kessaudit groups get privileged access to the application's functions.

Enter `yes` to remove all detected users from the kessadmin and/or kessaudit group. Users whose primary group is kessadmin or kessaudit are moved to the nogroup group. If there is no nogroup group, the installation will fail and you will be prompted to manually remove users from privileged groups.

Enter `no` if you do not want the application to remove users from the privileged groups.

Assigning the Administrator role to a user

At this step, you can grant the administrator (admin) role to the user.

Enter the name of the user to whom you want to grant the administrator role. You need the Administrator role to manage application settings and task settings in the graphical interface of the application and on the command line without using the `sudo` command.

You can assign the administrator role to a user later, after completing the initial configuration.

Determining the file operation interceptor type

At this step, the file operation interceptor type for the utilized operating system is determined. For operating systems that do not support fanotify technology, kernel module compilation will begin.

If all the required packages are available, the kernel module will be automatically compiled when the File Threat Protection task starts.

If, during the compilation of the kernel module, any dependencies are not found on the device, the Kaspersky Embedded Systems Security application suggests installing the relevant packages. If the package download fails, an error message will be displayed.

Enabling automatic configuration of SELinux

> This step is displayed only if SELinux is installed on your operating system.

At this step, you can enable automatic configuration of the SELinux access control system for working with Kaspersky Embedded Systems Security.

Enter `yes` to enable automatic configuration of SELinux. If SELinux cannot be configured automatically (for example, if the semanage utility is missing), the application displays an error message and prompts the user to configure SELinux manually.

Enter `no` if you do not want the application to automatically configure SELinux.

By default, the application suggests `yes`.

If necessary, you can manually configure SELinux to work with the application later, after the initial configuration of the application is complete.

Configuring the update source

At this step, you must specify the update source for databases and application modules. The application databases contain descriptions of the threat signatures and methods of countering them. The application uses these records when searching and neutralizing threats. Kaspersky virus analysts regularly add new records about threats.

Enter one of the following values:

- `KLServers`: the application receives updates from one of the Kaspersky update servers.
- `SCServer`: the application downloads updates to the protected device from Kaspersky Security Center Administration Server installed in your organization. You can select this update source if you use Kaspersky Security Center for centralized administration of device protection in your organization.
- `Custom`: the application downloads updates from a custom source. You can specify the address of the custom source of updates in the local area network or on the Internet.

Configuring proxy server settings

At this step, you must specify the proxy server settings if you are using a proxy server to access the Internet. Internet connection is required to download the application databases from the update servers.

To configure proxy server settings, perform one of the following actions:

- If you use a proxy server to connect to the Internet, specify the address of the proxy server using one of the following formats:
  - `<connection protocol>://<IP address of the proxy server>:<port number>`, if the proxy server connection does not require authentication
  - `<connection protocol>://<user name>:<password>@<IP address of the proxy server>:<port number>` if the proxy server connection requires authentication

    Connecting to a proxy server over HTTPS is not supported.

    When connecting via an HTTP proxy, we recommend to use a separate account that is not used to sign in to other systems. An HTTP proxy uses an insecure connection, and the account may be compromised.
- If you do not use a proxy server to connect to the Internet, enter no as your answer.

By default, the application suggests `no`.

You can manage proxy server settings later, after completing the initial configuration.

Starting an application database update

At this step, you can run the application database update task on the client device.

If you do not want to start to download the application databases, enter `no`.

If you want to start the database update task on the device, enter `yes`.

By default, the application suggests `yes`.

If `yes` is selected, the application will be automatically restarted after the databases are updated.

Kaspersky Embedded Systems Security protects the device only after the application databases are updated.

You can run the update task later, after completing the initial configuration.

Enabling automatic application database update

At this step, you can enable automatic update of the application databases.

Enter `yes` to enable automatic application database update. By default, the application checks for available database updates every 60 minutes. If updates are available, the application downloads the updated databases.

Enter `no` if you do not want the application to automatically update the databases.

You can enable automatic database update later, after completing the initial configuration, by configuring the update task schedule.

Activating the application

At this step, you can activate the application using an activation code or a key file.

To activate the application using an activation code, enter the activation code.

To activate the application using a key file, specify the full path to the key file.

If no activation code or key file is specified, the application is activated using a trial key for one month.

You can activate the application later, after completing the initial configuration.

Post-installation configuration of the application in automatic mode

To perform the initial configuration of the application in automatic mode:

1. Prepare a configuration file that contains the initial configuration settings. You can create this file or copy the necessary structure from the autoinstall.ini configuration file used for remote installation of the application using Kaspersky Security Center.

2. Pass the path to the configuration file to the Initial Configuration Wizard of the Kaspersky Embedded Systems Security application.

   You must run the Initial Configuration Wizard as root.

To launch the automatic initial configuration of the application, the following command:

```
# /opt/kaspersky/kess/bin/kess-setup.pl --autoinstall=<initial configuration file>
```

where <initial configuration file> is a path to the configuration file that contains initial configuration settings.

When the Initial Configuration Wizard finishes and releases the console, the post-installation configuration of the application is complete.

To check the return code, execute the following command:

```
echo $?
```

If the command returns 0, the post-installation configuration of the application is completed successfully.

Kaspersky Embedded Systems Security can protect the device only after the application databases are updated.

To correctly update application modules after the wizard has finished, the application may need to be restarted. Check the status of updates for the application using the `kess-control --app-info` command.

Settings in the configuration file for post-installation configuration

In the post-installation configuration file, you can specify the settings shown in the table below. The set of applicable settings depends on the application usage mode.

Settings in the configuration file for post-installation configuration

Setting	Description	Values
EULA_AGREED	Required setting. Acceptance of the terms of the End User License Agreement.	`yes` – accept the terms and conditions of the End User License Agreement. `no` – do not accept the terms and conditions of the End User License Agreement. The initial configuration of the application is aborted.
PRIVACY_POLICY_AGREED	Required setting. Acceptance of the terms of the Privacy Policy.	`yes`: accept the Privacy Policy to continue installing the application. `no` – do not accept the terms and conditions of the Privacy Policy. The initial configuration of the application is aborted.
USE_KSN	Required setting. Enabling Kaspersky Security Network usage: To enable the use of KSN, the terms of the Kaspersky Security Network Statement must be accepted.	`yes` – accept the terms of the Kaspersky Security Network Statement and enable the use of KSN. `no` – do not accept the Kaspersky Security Network Statement. Opting out of the KSN does not abort the initial configuration process. If KSN is enabled, the cloud mode is automatically enabled, in which Kaspersky Embedded Systems Security uses the lightweight version of malware databases.

Setting	Description	Values
GROUP_CLEAN	Required setting. Removing users from the kessadmin and kessaudit privileged groups.	`yes` - Remove users from the privileged groups. If the value is `yes` and there is no nogroup group, the installation will fail and you will be prompted to manually remove users from privileged groups. `no` - Do not remove users from the privileged groups.
LOCALE	Optional setting. The locale of the texts of the End User License Agreement, Privacy Policy, and Kaspersky Security Network Statement, as well as application events sent to Kaspersky Security Center.	The locale in the format specified by RFC 3066. If the `LOCALE` setting is not specified, the operating system locale is used. If the application fails to determine the operating system localization language or the operating system localization is not supported, the default value will be used – `en_US.utf8`. The locale of the graphical interface and the application command line depends on the value of the `LANG` environment variable. If the locale that is not supported by Kaspersky Embedded Systems Security is specified as the value of the `LANG` environment variable, the graphical interface and the command line are displayed in English.
INSTALL_LICENSE	Activation code or full path to the key file for adding a license key to the application.	`<activation code>` if you want to activate the application using an activation code during installation. `<full path to key file>` if you want to activate the application using a key file during installation. `None` if you do not want to activate the application during installation, for example, if you are installing the application in an isolated network segment without internet access.
UPDATER_SOURCE	Update source for databases and application modules.	`SCServer` – use the Kaspersky Security Center Administration Server as the update source. `KLServers` – use Kaspersky servers as the update source. `< update source address >`.
PROXY_SERVER	Address of the proxy server used to connect to the Internet.	Address of the proxy server in one of the following formats: • `[<connection protocol>://]<IP address of the proxy server>:<port>`, if no authentication is required to connect to the proxy server; • `[<connection protocol>://][<user name>:[<password>]@]<IP address of the proxy server>:<port>`, if authentication is required when connecting to the proxy server. Connecting to a proxy server over HTTPS is not supported.
UPDATE_EXECUTE	Start the application database update task during the initial configuration.	`yes` – start update task. `no` – do not start update task.
KERNEL_SRCS_INSTALL	Automatically start the compilation of the kernel module when the File Threat Protection task is started on operating systems that do not support the fanotify technology.	`yes` – compile the kernel module when the File Threat Protection task starts. `no` – do not compile the kernel module.
ADMIN_USER	A user assigned the administrator role (admin).	
CONFIGURE_SELINUX	Automatic configuration of SELinux for working with Kaspersky Embedded Systems Security.	`yes` – automatically configure SELinux to work with Kaspersky Embedded Systems Security. `no` – do not automatically configure SELinux to work with Kaspersky Embedded Systems Security.

Setting	Description	Values
DISABLE_PROTECTION	Disable protection components and scan tasks after the application is installed. Installation with protection components disabled can be convenient, for example, to reproduce a problem in the operation of the application and create a trace file. If you enable the necessary components and tasks after installing the application with the `DISABLE_PROTECTION=yes` parameter, the enabled components and tasks will continue to work after the application is restarted.	`yes` - Disable protection components and scan tasks when the application is started after installation. `no` - Do not disable protection components and scan tasks when the application is started after installation.
ENABLE_TRACES_ON_FIRST_STARTUP	Create a trace file when the application is started for the first time after installation. To create a trace, a kess_startup.log file is automatically created in the /var/log/kaspersky/kess/ directory. The size limit for regular application trace files does not apply to this file. Tracing stops after five minutes or after the File Threat Protection task is started.	`yes` – create a trace file when the application is started for the first time after installation. `no` – do not create a trace file when the application is started for the first time.

If you want to edit the settings in the configuration file for post-installation configuration of the application, specify the values of the settings in the <setting_name>=<setting_value> format (the application does not process spaces between the name of a setting and its value).

Configuring allowing rules in the SELinux system

If SELinux could not be configured automatically during the post-installation configuration of the application, or if you declined automatic configuration, you can manually configure SELinux to work with Kaspersky Embedded Systems Security.

To manually configure SELinux to work with the application:

1. Switch SELinux to permissive mode:
   - If SELinux has been activated, run the following command:

     ```
     # setenforce Permissive
     ```

   - If SELinux was disabled, set the `SELINUX=permissive` setting in the configuration file / etc / selinux / config and restart the operating system.
2. Make sure the semanage utility is installed on the system. If the utility is not installed, install the policycoreutils-python or policycoreutils-python-utils package, depending on the package manager.
3. If you are using a custom SELinux policy instead of the default targeted policy, assign a label to each source executable file of Kaspersky Embedded Systems Security in accordance with the SELinux policy being used; to do so, run the following commands:

   ```
   # semanage fcontext -a -t bin_t <executable file>
   # restorecon -v <executable file>
   ```

Where <executable file> is:

- /var/opt/kaspersky/kess/4.0.0.<build number>_<installation timestamp>/opt/kaspersky/kess/libexec/kess
- /var/opt/kaspersky/kess/4.0.0.<build number>_<installation timestamp>/opt/kaspersky/kess/bin/kess-control
- /var/opt/kaspersky/kess/4.0.0.<build number>_<installation timestamp>/opt/kaspersky/kess/libexec/kess-gui
- /var/opt/kaspersky/kess/4.0.0.<build number>_<installation timestamp>/opt/kaspersky/kess/shared/kess

4. Run the following tasks:

   - File Threat Protection task:

     ```
     kess-control --start-task 1
     ```

   - Critical Areas Scan task:

     ```
     kess-control --start-task 4 -W
     ```

   It is recommended to run all the tasks that you plan to run while using Kaspersky Embedded Systems Security.

5. Start the graphical user interface if you plan to use it.

6. Ensure that there are no errors in the audit.log file:

   ```
   # grep kess /var/log/audit/audit.log
   ```

7. If there are errors in the audit.log file, create and download a new rule module based on the blocking records in order to fix the errors, and then relaunch all the tasks that you plan to run while using Kaspersky Embedded Systems Security; to do so, run the following commands:

   ```
   # grep kess /var/log/audit/audit.log | audit2allow -M kess
   # semodule -i kess.pp
   ```

   If new audit messages related to Kaspersky Embedded Systems Security appear, the file with the rule module file must be updated.

8. Switch SELinux to blocking mode:

   ```
   # setenforce Enforcing
   ```

If you use a custom SELinux policy, manually assign a label to Kaspersky Embedded Systems Security source executable files after installing application updates (follow steps 1, 3–8).

You can find more information in the documentation for your operating system.

Running the application on Astra Linux OS in closed software environment mode

This section describes how to start the application in the Astra Linux Special Edition operating system in closed software environment mode.

For Astra Linux Special Edition (operational update 1.7) and Astra Linux Special Edition (operational update 1.8)

To start the application on the Astra Linux Special Edition operating system (update 1.7 or 1.8):

1. Specify the following setting in the /etc/digsig/digsig_initramfs.conf file:

   ```
   DIGSIG_ELF_MODE=1
   ```

2. Install the compatibility package:

   ```
   apt install astra-digsig-oldkeys
   ```

3. Create a directory for the application key:

   ```
   mkdir -p /etc/digsig/keys/legacy/kaspersky/
   ```

4. Locate the application key (/opt/kaspersky/kess/shared/kaspersky_astra_pub_key.gpg) in the directory created at the previous step:

   ```
   cp kaspersky_astra_pub_key.gpg /etc/digsig/keys/legacy/kaspersky/
   ```

5. Update the initramfs image:

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

6. Restart the operating system.

Running the application on ALT SP in closed software environment mode

This section describes how to start the application in the ALT SP operating system in closed software environment mode.

To run the application on the ALT SP operating system:

1. Disable the closed software environment mode, if enabled, by running the following command:

   ```
   integrity-remover
   ```

2. Create a kess group for further changes to the IMA policy by running the following command:

   ```
   groupadd kess
   ```

3. Find out the unique identifier (GUID) of the created group by running the following command:

   ```
   cat /etc/group | grep kess
   ```

4. Edit the IMA policy to exclude the kess group. To do so:

   a. Copy the /usr/share/integrity/policy file by running the following command:

      ```
      cp /usr/share/integrity/policy /etc/integrity/policy
      ```

   b. Edit the /etc/integrity/policy file using a text editor (for example, vim) by running the following command:

      ```
      vim /etc/integrity/policy
      ```

   c. After the `dont_measure fsmagic=0xf97cff8c` line (this is usually the last line in the `dont_measure` section), add the `dont_measure fgroup=<GUID>` line, where `<GUID>` is the unique ID of the group created at step 2 of these instructions).

   d. After the `dont_appraise fsmagic=0x27e0eb` line (this is usually the last line in the `dont_appraise` section), add the `dont_appraise fgroup=<GUID>` line, where `<GUID>` is the unique ID of the group created at step 2 of these instructions).

   e. Save your changes.

5. In the /usr/sbin/integrity-sign file, edit the list_all_files function to include the Kaspersky Embedded Systems Security application in the allowlist of applications that are allowed to run in the closed environment mode. To do so:

   a. Edit the /usr/sbin/integrity-sign file using a text editor (for example, vim) by running the following command:

      ```
      vim /usr/sbin/integrity-sign
      ```

   b. Find the list_all_files function and in the body of the function, in lines starting with `find -P`, add the directories of the Kaspersky Embedded Systems Security application under /var/opt and /opt to the end of each enumeration of directories.

   c. Save your changes.

6. Install the Kaspersky Embedded Systems Security application. To make sure that the application database update works correctly after the installation, at the initial configuration stage, refuse to start the application database update (to do this, enter 'no' at the application database update step).

7. Assign the inheritance of the kess group and directory permissions for the updatable components of the Kaspersky Embedded Systems Security application by running the following commands:

   ```
   cd /var/opt/kaspersky/kess/< build number _*>/var/opt/kaspersky/kess/private
   chown -R root:kess updates/
   find updates -type d -exec chmod g+s {} \;
   cd /opt/kaspersky/kess/lib64
   chown -R root:kess updatable_modules/
   find updatable_modules -type d -exec chmod g+s {} \;
   cd /var/opt/kaspersky/kess/common/
   chown -R root:kess temp/
   find temp -type d -exec chmod g+s {} \;
   ```

8. Assign the inheritance of the kess group and directory permissions for the updatable components of the Network Agent by running the following commands:

   ```
   cd /var/opt/kaspersky/
   chown -R root:kess klnagent/
   find klnagent/ -type d -exec chmod g+s {} \;
   cd /opt/kaspersky/
   chown -R root:kess klnagent64/
   find klnagent64/ -type d -exec chmod g+s {} \;
   ```

9. Enable the closed software environment mode by initiating the signing of the system using the following command:

   ```
   integrity-applier -i -A
   ```

10. Make sure that the IMA subsystem is activated by running the following command:

    ```
    cat /proc/cmdline
    ```

    The command output should contain the lines `lsm=integrity ima_hash=sha512 ima_appraise=enforce.`

11. Start the database update task by running the following command:

    ```
    kess-control --start 6 -W
    ```

    If the output contains the *EventType=BasesApplied* event, it means the database update was performed correctly.

12. Restart the application by running the following command:

    ```
    systemctl restart kess
    ```

Optimizing resource consumption by the application

This section contains recommendations of Kaspersky experts on configuring Kaspersky Embedded Systems Security to optimize system resource consumption. The application running with default settings may impact system performance on some devices because of the same resources being used by your applications and Kaspersky Embedded Systems Security. To avoid performance degradation, we recommend following the steps in this section after installing the application and completing the initial configuration.

We recommend the following steps to optimize system resource consumption by the application:

1. **Put your devices in administration groups according to the role of each device in the infrastructure.**

   Take an inventory of devices and the functions they perform and put them in different administration groups based on those functions. For example, you can have an administration group with web servers, another administration group with database servers, a third administration group with user workstations, and so on.

2. **Configure the application for typical server roles.**

   Configure the application on devices according to their role in the infrastructure as recommended.

3. **Analyze the performance on pilot devices.**

   Pick 1 or 2 pilot devices in each administration group. For each device, analyze how Kaspersky Embedded Systems Security uses system resources and how this impacts the performance of installed applications.

4. **Configure the application on pilot devices based on the results of performance analysis.**

   If the performance of applications is significantly degraded or the amount of system resources consumed by Kaspersky Embedded Systems Security is critical for the operation of the operating system and applications, configure File Threat Protection and scan tasks as recommended.

   You can also optimize the Web Threat Protection and Network Threat Protection components.

5. **Update the configuration of the application on all devices.**

   If the above steps helped improve performance on your pilot device, make the same changes on other devices of the same type.

Recommendations for typical server roles

Web server

When using HTTPS, with high client load on the server, the Web Threat Protection and Network Threat Protection components may significantly increase the consumption of system resources. This is caused by the application decrypting and scanning HTTPS traffic.

Analyze how the server load changes while these components are running. To get help choosing optimal parameters and resolve compatibility issues, please contact Kaspersky Technical Support.

Proxy server

We recommend disabling the Web Threat Protection and Network Threat Protection components on the device that acts as a proxy server or load balancer.

Server with containerization

We recommend disabling the Web Threat Protection and Network Threat Protection components on servers that are used for containerization services. To protect container environments, you can use the Kaspersky Container Security application.

Analysis of application performance on the device

When scanning objects, Kaspersky Embedded Systems Security uses CPU resources, disk subsystem input/output, and RAM.

To view the information about resource consumption by the application, execute the following command:

```
top -bn1|grep kess
```

The command must be executed when the system is under load.

The command output shows the amount of used memory and processor time:

```
651 root 20 0 3014172 2.302g 154360 S 120.0 30.0 0:32.80 kess
```

Column 6 displays the amount of resident memory – `2.302g`.

Column 9 displays the percentage of the processor cores usage – 120.0, where each core is represented by 100 percent. Thus, 120% means that one core is fully used, and the other is used at 20%.

If Kaspersky Embedded Systems Security critically slows down your system, you need to find out which application components and tasks are taking up system resources. To do this, examine the following on the device using the command line:

1. Analyze the resource consumption of the File Threat Protection task (task of the OAS type). In this way, you can find out if the performance issues are related to the File Threat Protection task or the scan tasks (tasks of the ODS and ContainerScan types), and follow the recommendations for the relevant tasks.

2. Identify the most frequently scanned objects. You can optimize performance by excluding safe objects and files from scanning.

You can analyze the performance of tasks on the command line. If the device is managed by Kaspersky Security Center policy, you need to allow local task management while you carry out your investigation.

Analyzing the resource consumption of the File Threat Protection task

To analyze the resource consumption of the File Threat Protection task:

1. Stop all scan and monitoring tasks.
2. Make sure that on-demand scan tasks (tasks of the ODS and ContainerScan types) are not scheduled to run during your investigation:
   a. Get the list of all application tasks by executing the following command:
      `kess-control --get-task-list`
   b. Get the schedule settings for all tasks of the ODS and ContainerScan types by running the following command:
      `kess-control --get-schedule <task ID/name>`
      If the command output is `RuleType=Manual`, the task can only be started manually.
   c. For all tasks that have a `RuleType` other than `Manual`, change the schedule by running the following command:
      `kess-control --set-schedule <task ID/name> RuleType=Manual`
3. Enable generation of application trace files with a high level of details by executing the following command:
   `kess-control --set-app-settings TraceLevel=Detailed`
4. If the File Threat Protection task is not running, start it with the following command:
   `kess-control --start-task 1`
5. Load the system in the mode that caused the performance problems; a few hours is enough.
6. Analyze the resulting trace files. By default, trace files are saved in the /var/log/kaspersky/kess/ directory.

   Under load, the application writes a lot of information to the trace files; however only 10 files 500 MB each are kept, meaning that older information is overwritten.

   If performance and resource consumption issues persist, you may need to optimize File Threat Protection.

   If performance and resource consumption issues no longer occur, then the most likely cause of the issues are the on-demand scan tasks (tasks of the ODS and ContainerScan types). This means you need to optimize these tasks.
7. Disable creation of the application trace files by executing the following command:
   `kess-control --set-app-settings TraceLevel=None`

Identifying the most frequently scanned objects

You can get the list of objects that have been scanned the most times by running the following command:

```
fgrep 'AVP ENTER' /var/log/kaspersky/kess/kess.* | awk '{print $8}' | sort | uniq -c |
sort -k1 -n -r|less
```

The result is loaded into less, a text viewer utility, where the objects that have been scanned the most times are displayed first.

If you need to aggregate object scanning statistics in subdirectories down to a certain depth starting from the root directory `/`, you can add `cut -d/ -f 1-X` to the command, where X is the nesting depth, for example:

```
fgrep 'AVP ENTER' /var/log/kaspersky/kess/kess.* | awk '{print $8}' | cut -d/ -f 1-4 |
sort | uniq -c | sort -k1 -n -r|less
```

If various services frequently write data to files in the system, such files are scanned again in the pending queue. Get the list of paths that have been scanned the most times in the pending queue by running the following command:

```
fgrep 'SYSCALL' /var/log/kaspersky/kess/kess.* | fgrep 'KLIF_ACTION_CLOSE_MODIFY' | awk
'{print $10}' | sort | uniq -c | sort -k1 -n -r
```

The files that were scanned the most times will appear at the beginning of the list.

If you need to aggregate object scanning statistics in subdirectories down to a certain depth starting from the root directory `/`, you can add `cut -d/ -f 1-X` to the command, where X is the nesting depth, for example:

```
fgrep 'SYSCALL' /var/log/kaspersky/kess/kess.* | fgrep 'KLIF_ACTION_CLOSE_MODIFY' | awk
'{print $10}' | tr -d ',' | cut -d/ -f 1-3 | sort | uniq -c | sort -k1 -n -r
```

We recommend determining whether the objects scanned the most number of times are dangerous. You can add objects and files confirmed as safe to exclusions to optimize File Threat Protection and scan tasks. For example, database files, directories, and log files can be considered safe if a write operation is performed by a trusted process. In case of any difficulties, contact Technical Support.

Recommendations for File Threat Protection and scan tasks

Optimizing File Threat Protection

To optimize the File Threat Protection component, you can:

- Configure scan exclusions for objects that are scanned most frequently and are safe.
- Configure exclusions from file operation interception.

- Change the file scanning mode to scan a file only when the file is opened. By default, the application scans a file when something attempts to open it, and scan it again when something attempts to close it, if the file was modified. You can change the file scanning mode in the File Threat Protection settings:

  - In the policy using the Web Console or Administration Console: set **File Threat Protection mode** to **When opened**.

  - Locally on the device using the following command:

    ```
    kess-control --set-set 1 ScanByAccessType=Open
    ```

  Changes made to the file after it is opened are not scanned until the next opening of the file.

- Make sure that archive scanning is disabled (default value). Scanning archives impacts performance because access to the archive file is blocked while the file is being unpacked to the /tmp temporary directory and scanned. The definition of archive includes traditional archives as well as packaged objects such as .jar.

  You can enable or disable archive scanning using the Web Console or Administration Console (the **Scan archives** setting) or locally on the device using the command line (the `ScanArchived` setting).

- Limit the amount of RAM the application can use.

Optimizing scan tasks

To optimize scan tasks (tasks of the ODS and ContainerScan types), you can:

- Schedule scan tasks so that they do not overlap. The application allows running scan tasks simultaneously, but this may significantly increase resource consumption. Check the schedule of all ODS and ContainerScan tasks locally on the device or in Kaspersky Security Center.

- Run the scan at a time when the server load is minimal.

- Make sure that the scope of the scan tasks does not include remote directories mounted on the device via SMB and NFS. We recommend scanning such directories directly on the device that provides the resource. If this is not possible, do not perform the scan on servers that run critical services, because such a task may take quite a long time (depending on the connection speed and the number of files).

- Configure scan exclusions for objects that are scanned most frequently and are safe.

- Configure exclusions from file operation interception.

- Limit the amount of RAM the application can use.

Configuring scan exclusions

Scan exclusions help balance system security with performance. For example, according to Kaspersky virus experts, you can safely add data files and index files of databases to exclusions. It is believed that no danger exists of such files becoming virus vectors.

You can exclude safe objects on the list of most frequently scanned objects.

You can configure exclusions by file path (you can also specify the mount type), file mask, threat type, and path to the process performing file operations.

The application does not scan objects excluded from scanning, but file operations still experience minimal delays because the application still performs file operation interception and blocking. If you need to prevent even minimal delays, we recommend configuring exclusions from file operation interception.

You can configure File Threat Protection exclusions in a policy using the Web Console or Administration Console, or locally on the device using the command line.

You can also configure exclusions for scan tasks of the ODS and ContainerScan types. Exclusions are configured in task settings in Kaspersky Security Center or locally on the device using the command line.

Scan exclusion settings for one scan task do not affect other scan tasks. Exclusions must be configured separately for each scan task.

Below are examples of configuring File Threat Protection exclusions on the command line.

To exclude the /tmp/logs directory and all subdirectories and files recursively, execute the following command:

```
kess-control --set-settings 1 --add-exclusion /tmp/logs
```

To exclude a specific file or files by mask in the /tmp/logs directory, execute the following command:

```
kess-control --set-settings 1 --add-exclusion /tmp/logs/*.log
```

To exclude all files with the .log extension in the /tmp/ directory and subdirectories using a recursive mask, execute the following command:

```
kess-control --set-settings 1 --add-exclusion /tmp/**/*.log
```

Configuring exclusions from file operation interception

Files excluded from scanning may still be intercepted by the application during File Threat Protection and scanning tasks. The application intercepts file operation events and blocks actions until the events are processed. After the scan, the application can:

- Unblock and allow the operation.
- Block the operation and perform the actions specified in protection or verification settings.

Intercepting events takes time, which affects the performance of other applications and the operating system.

You can get file interception statistics using the following command:

```
kess-control --get-statistic --files
```

This command lets you display file operation interception statistics over the last 24 hours.

If necessary, you can use application trace files for more detailed analysis. By default, the application does not generate trace files. You can enable tracing using the following command:

```
kess-control --set-app-settings TraceLevel=Detailed
```

Enabling tracing may impact overall system performance. We recommend disabling tracing when you no longer need trace files.

You can disable tracing using the following command:

```
kess-control --set-app-settings TraceLevel=None
```

By analyzing trace files, you can get a list of files intercepted by the application and identify paths associated with a lot of file operation interception. You can get a list of intercepted files by running the following command:

```
grep 'FACACHE.*needs' /var/log/kaspersky/kess/kess.* | awk '{print $9}' | tr -d '}'|awk -F':' '{print $2}'|sort | uniq -c | sort -k1 -n -r
```

If you need to aggregate file operation interception statistics in subdirectories down to a certain depth starting from the root directory `/`, you can add `cut -d/ -f 1-X` to the command, where X is the nesting depth, for example:

```
grep 'FACACHE.*needs' /var/log/kaspersky/kess/kess.* | awk '{print $9}' | tr -d '}'|awk -F':' '{print $2}'| cut -d/ -f 1-3|sort | uniq -c | sort -k1 -n -r|less
```

To exclude files in a certain directory not only from scan, but also from interception, you need to exclude the entire mount point. Exclusion of mount points (global exclusion) allows you to exclude local or remote directories mounted on the device from interception of file operations.

Global exclusions have no effect on paths that are not mount points.

Kaspersky experts recommend using global exclusions in the following cases:

- The connection to a mounted remote resource, such as NFS or SMB, is slow.
- Even minimal lag is not acceptable.

To protect shared network resources, use the Kaspersky Embedded Systems Security application on the device where the resources are physically stored.

You can configure global exclusions in a policy using the Web Console or Administration Console, or locally on the device using the command line.

Below is an example of configuring a file operation interception exclusion using the command line.

To exclude all files in the /tmp directory from interception:

1. If the directory is not a mount point, create a mount point from it. For example, to create a mount point from the /tmp directory, run the following command:

   ```
   mount --bind /tmp/ /tmp
   ```

2. To keep the mount point after the server reboot, add the following line to the /etc/fstab file:

   ```
   /tmp /tmp none defaults,bind 0 0
   ```

3. Add the /tmp directory to the global exceptions by executing the following command:

   ```
   kess-control --set-app-settings ExcludedMountPoint.item_0000=/tmp
   ```

4. If you want to add several directories, increase the item_0000 counter by one (item_0001, item_0002, and so on).

Setting the application memory usage limit

You can limit the amount of RAM that Kaspersky Embedded Systems Security uses when running OAS and ODS tasks.

Limiting memory usage can be useful for systems with a large amount of RAM (more than 5 GB).

You can use the `ScanMemoryLimit` option in the kess.ini configuration file to adjust the size of RAM used by the application when scanning files. Default value: 8192 MB. The minimum value is 2 MB. If the specified value is less than 2 MB, the application uses the minimum value (2 MB). If the specified value is greater than the amount of RAM available on the device, the application uses up to 25% of the RAM. This value cannot be changed.

> This setting limits only the amount of memory used when scanning files. That means that the total amount of memory required by the application can be more than the value of this setting.

To specify a limit on memory use when scanning files:

1. Stop Kaspersky Embedded Systems Security.
2. Open the /var/opt/kaspersky/kess/common/kess.ini file for editing.
3. Under `[General]`, specify the required amount of RAM in megabytes in the `ScanMemoryLimit` setting:

   ```
   ScanMemoryLimit=< amount of memory in megabytes >
   ```

4. Start Kaspersky Embedded Systems Security.

   The new memory usage limit for scanning files will be in effect after the application restarts.

Recommendations for the Web Threat Protection and Network Threat Protection tasks

To reduce the impact of the Web Threat Protection and Network Threat Protection components on the performance of your applications, we recommend taking the following steps.

- Configure the firewall of the operating system. If the system has its own network rules (or firewalld rules) with Drop policies, then for the application to work correctly, you need to add an accept rule to allow incoming connections from the local device (loopback):

  Example command for iptables:

```
iptables -A INPUT -i lo -j ACCEPT
```

- To ensure the compatibility of a REDIRECT rule (for example, redirection from port 443 to 8443) with the Network Threat Protection component interception rules:

  a. Delete a rule such as the following:

  ```
  iptables -t nat -A PREROUTING -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 8443
  ```

  b. Replace it with the following rules:

  ```
  iptables -t nat -A PREROUTING -p tcp -m tcp --dport 443 -m mark ! --mark 0x400/0x400
  -j REDIRECT --to-ports 8443
  ```

  ```
  iptables -t nat -A OUTPUT -p tcp --dport 443 -m mark --mark 0x400/0x400 -m addrtype
  ! --src-type LOCAL -j DNAT --to-destination :8443
  ```

- Configure certificate verification policy. By default, the Web Threat Protection component decrypts and scans network traffic sent over encrypted connections. When doing so, the component can use the internet for the verification and download missing chains required to verify the certificate. If the device does not have direct internet access, you must select a local certificate verification method (without using the internet) or use a proxy server for internet access.

  You can select a local certificate verification method in the network traffic scan settings:

  - In the policy using the Web Console or Administration Console: set **Certificate verification policy** to **Local check**.

  - Locally on the device using the following command:

    ```
    kess-control --set-net-settings CertificateVerificationPolicy=LocalCheck
    ```

  The application will not use the internet to verify certificates.

  You can configure a proxy server in a policy using the Web Console, Administration Console, or locally on the command line.

- Limit the list of network ports monitored by the application. By default, the application only monitors selected network ports. If the device uses an SMB connection, make sure the default value is used:

  - In the policy in the Web Console or Administration Console: in the network traffic scan settings, the **Monitor selected network ports only** option is selected.

  - Locally on the device: the `MonitorNetworkPort` network traffic scan setting is set to `Selected`.

When using the application together with the HAProxy load balancer, we recommend taking the following steps:

1. Open the /var/opt/kaspersky/kess/common/kess.ini configuration file, create the `[Environment]` section and add the `TcpSynInterceptDisabled=1` setting to it.

2. Restart the application:

   ```
   systemctl restart kess
   ```

Using the Web Threat Protection and Network Threat Protection components together with Kubernetes containerization requires additional painstaking fine-tuning of the operating system firewall. The components cannot be used with Kubernetes if Cillium CNI is also used.

Updating the application from a previous version

You can update Kaspersky Embedded Systems Security 3.4 for Linux to Kaspersky Embedded Systems Security 4.0 for Linux.

You need to prepare for installation before updating Kaspersky Embedded Systems Security.

The application update procedure involves the following steps:

1. **Updating the Kaspersky Security Center Network Agent**

   If you are managing Kaspersky Embedded Systems Security using Kaspersky Security Center, you must update the Network Agent on the protected devices. The update is performed by installing a new version of Network Agent.

   If the Network Agent is not updated, the application cannot be managed using Kaspersky Security Center.

   > On a device running the Astra Linux Special Edition operating system, we recommend to update Network Agent remotely using Kaspersky Security Center, since updating using the command line in the Kaspersky Security Center administration console creates a new copy of the same managed device, and the old one becomes inaccessible.

   The application continues working correctly during the Network Agent update.

2. **Updating the Kaspersky Embedded Systems Security management plug-in**

   If you are managing Kaspersky Embedded Systems Security using Kaspersky Security Center, you must update the Kaspersky Embedded Systems Security administration web plug-in or MMC plug-in, depending on the console that you are using to manage Kaspersky Security Center.

3. **Updating the application and graphical user interface on protected devices**

   You must update the application installed on protected devices. If you are using the application GUI, you also need to update the GUI.

   You can update the application and the application's graphical user interface in the following ways:

   - Remotely using Kaspersky Security Center.
   - Locally from the command line.

> If an error occurs while updating the application, the update is rolled back and the previous version of the application is started. In this case, an error message will be displayed, but the package manager (rpm/dpkg) will indicate the new version.

Even if Kaspersky Embedded Systems Security is launched before the update process start, if the update is completed successfully, a new application version is launched.

When you update the application to a newer version, the dump files of the previous version are deleted.

After updating the application, it is recommended to run the database update task.

About updating Kaspersky Embedded Systems Security management plug-ins

The management plug-in for Kaspersky Embedded Systems Security is updated by installing the new version of the management plug-in. Depending on the Kaspersky Security Center administration console that you use, you have to install:

- Kaspersky Embedded Systems Security administration web plug-in
- Kaspersky Embedded Systems Security administration MMC plug-in

Policies and tasks configured for previous versions of Kaspersky Embedded Systems Security are not compatible with the updated version of the application. If you use the Kaspersky Security Center Administration Console to manage the application, then after updating the management MMC plug-in, you can convert policies and tasks using the Kaspersky Security Center Policies and Tasks Batch Conversion Wizard (see more details in the Kaspersky Security Center Help). The converted policies and tasks have names "<Original policy/task name> (converted)". For most settings, converted policies and tasks use the values configured for the previous version of the application. Some settings are assigned special values. The settings that were not configured in the policies and tasks of the previous version take default values in the converted policies and tasks.

The procedure for converting policies and tasks is not available in Kaspersky Security Center Web Console. If you use the Web Console to manage the application, you must create new policies and tasks for the application in Kaspersky Security Center.

Exporting and importing policy settings from previous versions in the Web Console is not supported. You can migrate some task settings from a previous version of the task to a new one by exporting and importing settings.

Management plug-ins of the previous version continue to work after installing the new version of Kaspersky Embedded Systems Security management plug-ins. You can use them to manage the previous version of Kaspersky Embedded Systems Security.

If you have updated the application on all client devices, you can uninstall the Kaspersky Embedded Systems Security management plug-ins of the previous version.

When installing a new build of the MMC management plug-in for the same version of the application, you must first uninstall the currently installed MMC plug-in of this version, and then install the new MMC plug-in of this version.

Updating the application using Kaspersky Security Center

The application and graphical user interface are updated by remotely installing the new version of the application packages and graphical user interface on the protected device.

Updating using Kaspersky Security Center involves the following steps:

1. Creating an installation package.

   For the remote installation, Kaspersky Embedded Systems Security installation package is used. You can create the installation package using the Kaspersky Security Center Web Console or the Administration Console.

2. Deploying the Kaspersky Endpoint Security application on devices in the corporate network.

   Kaspersky Security Center Web Console supports the following main deployment methods:

   - Installing the application using the Protection Deployment Wizard.
   - Installing the application using the remote installation task.

   The Kaspersky Security Center Administration Console supports the following main deployment methods:

   - Installing the application using the Remote Installation Wizard.
   - Installing the application using the remote installation task.

   For a description of the deployment procedures, see the Kaspersky Security Center Help.

Updating the application using the command line

Updating the application using the command line is performed by installing a new version of the application on the device from an RPM or DEB format package depending on the type of package manager.

If the terms and conditions of the End User License Agreement and/or the Privacy Policy have changed in the new version of the application, you must accept the new terms and conditions during the update. Read the new version of the End User License Agreement and/or the Privacy Policy:

- The new version of the End User License Agreement is located in the (~/.kess/<application version>/license.<language ID>) directory.
- The new version of the Privacy Policy is located in the (~/.kess/<application version>/license.<language ID>) directory.

If you do not accept the conditions of the End User License Agreement and/or the Privacy Policy, the application will not be updated.

If the terms of the Kaspersky Security Network Statement changed in the new version of the application, you need to accept or decline the new terms of use for participating in Kaspersky Security Network. Read the new version of the document located in the (~/.kess/<application version>/ksn_license.<language ID>) directory. Refusing to use Kaspersky Security Network will not halt the Kaspersky Embedded Systems Security update process. You can enable, disable, or change Kaspersky Security Network mode later.

If you used KSN and accepted the conditions of the Kaspersky Security Network Statement in a previous version of the application, you need to accept the conditions of the Kaspersky Security Network Statement when updating the application. Otherwise, use of KSN will be disabled.

To accept the terms of the new agreements during the upgrade, use the variables `KESS_EULA_AGREED=yes`, `KESS_PRIVACY_POLICY_AGREED=yes`, and `KESS_USE_KSN=yes/no`.

To update the application:

1. Install the application package using the following command, depending on the package manager. If you have the graphical user interface of the previous version of the application installed, then you also need to start the package containing the files of the graphical user interface.

   - for an RPM package:

     `# [KESS_EULA_AGREED=yes] [KESS_PRIVACY_POLICY_AGREED=yes] [KESS_USE_KSN=yes/no] rpm -U --replacefiles --replacepkgs kess-4.0.0-<build number>.<arch>.rpm [kess-gui-4.0.0-<build number>.<arch>.rpm]`

     where `<arch>` is the architecture type:

     - i386 – for 32-bit operating systems
     - x86_64 – for 64-bit operating systems
     - aarch64 – for 64-bit operating systems for the Arm architecture

     On an operating system with a package manager of the RPM type, if the application package and the GUI package are both installed, we do not recommend updating only one of the packages without the other.

   - for a DEB package:

     `# [KESS_EULA_AGREED=yes] [KESS_PRIVACY_POLICY_AGREED=yes] [KESS_USE_KSN=yes/no] apt-get install ./kess_4.0.0-<build number>_<arch>.deb [./kess-gui_4.0.0-<build number>_<arch>.deb]`

     where `<arch>` is the architecture type:

     - i386 – for 32-bit operating systems
     - amd64 – for 64-bit operating systems
     - arm64 – for 64-bit operating systems for the Arm architecture

     On an operating system with a package manager of the dpkg type, if the application package and the GUI package are both installed, either of the packages cannot be updated without the other.

2. Kaspersky Embedded Systems Security restarts automatically.

Example:
Install the RPM application package with the GUI package for 64-bit operating systems*:*

`# KESS_EULA_AGREED=yes KESS_PRIVACY_POLICY_AGREED=yes KESS_USE_KSN=yes rpm -U --replacefiles --replacepkgs kess-4.0.0-<build number>.x86_64.rpm kess-gui-4.0.0-<build number>.x86_64.rpm`

If you use the command line to manage the application, then after upgrading, most application settings use the values configured for the previous version of the application. Some settings are assigned special values. Settings that were missing in the previous version of the application take on default values in the new version of the application.

Changes to the application settings made after the update is complete and before the application restarts are not saved.

Special considerations when setting parameter values when updating the application

If you use Kaspersky Security Center Administration Console to manage the application, you can convert policies and tasks to use the values of policy and task settings configured for the previous version of the application (for more information, see the Kaspersky Security Center Help). The procedure for converting policies and tasks is not available in Kaspersky Security Center Web Console.

After updating the application using the command line, most settings carry over from the previous version of the application. You can also migrate application settings by exporting settings to a file and then importing them from that file.

Default values are assigned to settings that did not exist in the previous version of the application. Some settings are assigned special values.

Kaspersky Security Network settings

After converting a policy in the MMC plug-in, the Kaspersky Security Network settings in the policy properties depend on whether you accepted or rejected the terms of the Kaspersky Security Network Statement in the Policies and Tasks Batch Conversion Wizard:

- If you accepted the terms of the Statement, the **Extended KSN mode** option is selected.
- If you rejected the terms of the Statement, the **Do not use KSN** option is selected.

The conversion of policies is not supported by the web plug-in.

After upgrading the application on the command line, the `UseKSN` setting is set to `No` if when updating you set `KESL_USE_KSN=No`, and `UseKSN=Extended` is applied if you set `KESL_USE_KSN=Yes`.

Cloud mode settings

After converting a policy in the MMC plug-in, the settings for using cloud mode may change depending on whether you accepted or rejected the terms of the Kaspersky Security Network Statement in the Policies and Tasks Batch Conversion Wizard:

- If you accepted the terms of the Statement, the **Enable cloud mode** check box will be selected.
- If you rejected the terms of the Statement, the **Enable cloud mode** check box will be cleared.

The conversion of policies is not supported by the web plug-in.

After updating the application on the command line, `CloudMode` is set to:

- `CloudMode=No` if `UseKSN=No` after the upgrade.
- `CloudMode=Yes` if `UseKSN=Basic/Extended` was set after the update and `CloudMode=Yes` was set before the update.

Cloud mode is available if use of KSN is enabled. To enable cloud mode:

- If using the MMC or web plug-in, select the **Extended KSN mode** option and select the **Enable cloud mode** check box.
- When using the command line, set the `UseKSN` parameter to `Basic` or `Extended` and the `CloudMode` parameter to `Yes`.

Uninstalling the application

Uninstalling the Kaspersky Embedded Systems Security involves the following steps:

1. **Uninstalling the application and graphical interface of the application**

   Uninstall the packages of the application and, if you are using the graphical interface, the packages of the graphical interface from the protected devices.

   You can uninstall both the application package and the graphical interface package, or uninstall only the graphical interface package. It is not possible to uninstall only the application package if the graphical interface package is installed.

   You can uninstall the application and the application's graphical interface in the following ways:

   - Remotely using Kaspersky Security Center.
   - Locally from the command line.

   > While the application is being uninstalled, all Kaspersky Embedded Systems Security tasks will be stopped on the device.

2. **Removing Network Agent**

   If you were using Kaspersky Security Center to manage Kaspersky Embedded Systems Security, you must uninstall the Network Agent from protected devices.

   You can uninstall Network Agent in the following ways:

   - Remotely using Kaspersky Security Center.
   - Locally from the command line.

3. **Installing the Kaspersky Embedded Systems Security management plug-in**

   If you were using Kaspersky Security Center to manage Kaspersky Embedded Systems Security, you must uninstall the Kaspersky Embedded Systems Security administration web plug-in or MMC plug-in, depending on the console that you were using to manage Kaspersky Security Center.

> After removing the application, all information saved by the application is deleted, except for the license database. Installed application certificates are also removed. The license database is saved, and you can use it to reinstall the application.

If the application was installed in a systemd, the systemd settings are restored to their initial state after the application uninstallation.

Uninstalling the application and Network Agent using Kaspersky Security Center

You can remotely uninstall Kaspersky Embedded Systems Security and Network Agent from the client devices.

Uninstallation is performed using the remote uninstallation of applications task in Kaspersky Security Center Web Console or in the Administration Console. For more information, see the Kaspersky Security Center Help.

If you want to remove only the graphical user interface without removing the application, specify the `USE_GUI=No` setting value in the autoinstall.ini configuration file and start the remote application installation task.

Uninstallation is performed in the background. After the application uninstallation finishes, you will be prompted to restart the client device.

Uninstalling the application using the command line

Removing the application package and the graphical user interface package

To uninstall the application and the graphical interface installed from the RPM packages, run the following command:

```
# rpm -e kess kess-gui
```

To remove the application and the graphical user interface installed from DEB packages, run the following command:

```
# apt-get purge kess kess-gui
```

Uninstalling the application package without the graphical user interface package

To uninstall the application installed from the RPM package without removing the graphical user interface, run the following command:

```
# rpm -e kess
```

To uninstall the application installed from the DEB package without removing the graphical user interface, run the following command:

```
# apt-get purge kess
```

Removing the graphical user interface package

To remove the graphical interface that was installed from an RPM package, run the following command:

```
# rpm -e kess-gui
```

To remove the graphical interface that was installed from a DEB package, run the following command:

```
# apt-get purge kess-gui
```

After the uninstallation procedure is complete, a message about the results of the uninstallation is displayed.

Network Agent removal using the command line

To uninstall the Network Agent installed on a 32-bit operating system from an RPM package, run the following command:

```
# rpm -e klnagent
```

To uninstall the Network Agent installed on a 64-bit operating system from an RPM package, run the following command:

```
# rpm -e klnagent64
```

To uninstall the Network Agent installed on a 32-bit operating system from a DEB package, run the following command:

```
# apt-get purge klnagent
```

To uninstall the Network Agent installed on a 64-bit operating system from a DEB package, run the following command:

```
# apt-get purge klnagent64
```

After the uninstallation procedure is complete, a message about the results of the uninstallation is displayed.

Removing Kaspersky Embedded Systems Security management plug-ins

The Kaspersky Embedded Systems Security administration web plug-in is uninstalled in Kaspersky Security Center Web Console from the list of installed plug-ins (**Settings** → **Web Plug-ins**).

To remove the Kaspersky Embedded Systems Security administration MMC plug-in:

1. Close the Kaspersky Security Center Administration Console.
2. On the device where the Kaspersky Security Center Administration Console is installed, open the Windows registry editor and go to:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\KasperskyLab\Components\28\Plugins
   ```

   This key contains information about all management plug-ins installed in the Administration Console. The name of the managed application is in the **DisplayName** value.

3. Select the section corresponding to the relevant version of the Kaspersky Embedded Systems Security application plug-in.

4. Open and copy the **UninstallString** value.

5. Open the command prompt as administrator, paste the copied value and press **Enter**.

Application licensing

This section contains information about the basic concepts associated with licensing Kaspersky applications, as well as information about the specifics of activating the Kaspersky Embedded Systems Security application.

After activating the application, we recommend monitoring the license validity period in order to renew the license in a timely manner when necessary. You can use Kaspersky Security Center or the command line on a protected device to view information about license keys used by Kaspersky Embedded Systems Security.

About the End User License Agreement

The *End User License Agreement* is a binding agreement between you and AO Kaspersky Lab, stipulating the terms on which you may use the application.

Read through the terms of the End User License Agreement carefully before you start using the application.

You can read the End User License Agreement and the Privacy Policy, which describes data processing and provision:

- Before installing the application in the license.<language identifier> file. This file is included in the application distribution kit.
- During Kaspersky Embedded Systems Security installation.

  By confirming your consent to the text of the End User License Agreement and Privacy Policy when creating the application installation package (if installed using Kaspersky Security Center) or during the initial application configuration (if installing using the command line), you accept the terms of the End User License Agreement and Privacy Policy If you do not accept the terms of the End User License Agreement or Privacy Policy, you must cancel the installation of the application and may not use the application.
- After the application is installed, the files containing the texts of the End User License Agreement of the Kaspersky application and the Privacy Policy are located on the protected device in the /opt/kaspersky/kess/doc/license.<language ID> directory.

About the license

A *license* is a time-limited right to use Kaspersky Embedded Systems Security, granted under the End User License Agreement.

The list of available functions and the validity period of the application depend on the license under which the application is used.

The following license types are provided:

- *Trial* – a free license intended for trying out the application.

Trial licenses have a short validity period. When the trial license expires, all Kaspersky Embedded Systems Security features become disabled. To continue using the application, you need to purchase a commercial license.

You can use the application under a trial license for only one trial period.

- *Commercial* is a paid license.

  The main functions of the application stop working when a commercial license expires. To continue using Kaspersky Embedded Systems Security, you need to renew the commercial license. After the license expires, you can no longer use the application and must uninstall it from the device.

  It is recommended to renew the license before its expiration date to ensure continued protection of your device against security threats.

About license certificate

A *license certificate* is a document transferred to the user together with a key file or activation code.

The license certificate contains the following license information:

- License ID or order number.
- Details of the user to whom the license is granted.
- Details of the application that can be activated using the license.
- Limitation on the number of licensed units (for example, the number of devices on which the application can be used under the license).
- License term start date.
- License expiration date or license term.
- License type.

About license key

A *license key* is a sequence of bits that you can use to activate and then use the application in accordance with the terms of the End User License Agreement. License key is generated by Kaspersky experts.

To activate the application, you need the main license key. A separate license key is provided for the Kaspersky Embedded Systems Security application that can be used to activate the integration with Kaspersky Managed Detection and Response; this is called the MDR key.

You can add a license key to the application using one of the following methods: by applying a *key file* or by entering an *activation code*. After you add a key to the protected device, the license key is displayed in the application interface as a unique alphanumeric sequence.

The license key may be blocked by Kaspersky, if the terms of the End User License Agreement are violated. If the license key is blocked, add another license key for proper application operation.

A license key can be added as an active key or as a reserve key:

- *Active license key* is currently used to run the application. A trial license key or a commercial license key (commercial key) can be added as the active key. You can add only one main license key and one MDR license key as an active key.

- *Reserve license key* is a license key that entitles the user to use the application, but is not currently in use. The reserve license key automatically becomes active when the license associated with the current active license key expires.

  To extend the term of the application's main license, you can add a reserve key for the basic functionality. If you have activated additional functionality under a separate license, you can also add a reserve key for additional functionality.

  A reserve license key can be added only if an active license key of the same type is already added.

A trial license key can only be added as an active license key.

About activation code

An *activation code* is a unique sequence of 20 alphanumeric characters. You have to enter an activation code in order to add a license key for activating Kaspersky Embedded Systems Security. You receive the activation code at the email address that you provided when you bought Kaspersky Embedded Systems Security or requested the trial version of Kaspersky Embedded Systems Security.

To activate the application with an activation code, you need Internet access in order to connect to Kaspersky activation servers.

If you lost your activation code after activating the application, contact the Kaspersky partner from whom you purchased the license.

About the key file

A *key file* is a file with the .key extension that you receive from Kaspersky. The purpose of a key file is to add a license key that activates the application.

You receive a key file at the email address that you provided when you bought Kaspersky Embedded Systems Security or ordered the trial version of Kaspersky Embedded Systems Security.

You do not need to connect to Kaspersky activation servers in order to activate the application with a key file.

You can restore a key file if it has been accidentally deleted. You may need a key file to register a Kaspersky CompanyAccount, for example.

To restore your key file, perform any of the following actions:

- Contact the license seller.

- Obtain a key file on the Kaspersky website based on your existing activation code.

About subscription

Subscription for Kaspersky Embedded Systems Security is a purchase order for the application with specific settings (subscription expiry date, number of devices protected). You can order a subscription for Kaspersky Embedded Systems Security from your service provider (such as your internet service provider). You can renew or cancel your subscription. You can manage your subscription on the website of the service provider.

Subscription can be limited (for one year, for example) or unlimited (without an expiry date). To continue using the application after the limited subscription expires, you need to renew your subscription. Unlimited subscription is renewed automatically if the vendor's services have been prepaid on time.

Upon a limited subscription's expiry, you may be offered a grace period to renew the subscription. During this period the application retains its functionality. The service provider decides whether or not to grant a grace period and, if so, determines the duration of the grace period.

The set of options for managing your subscription may vary depending on your service provider. The service provider might not provide a grace period for renewing the subscription where the application retains its functionality.

To use Kaspersky Embedded Systems Security under a subscription, you need to use the activation code received from the service provider. After you apply the activation code, an active key corresponding to the license to use the application under subscription is added to the application. A reserve key can only be added when you use an activation code and cannot be added for a key file or subscription.

Activation codes purchased under subscription may not be used to activate previous versions of Kaspersky Embedded Systems Security.

Application activation and license key management

Activation is the process of activating an application using a license that allows you to use a fully functional version of the application until the license expires.

To activate the Kaspersky Embedded Systems Security application on a protected device, you need to add a license key to the application. Also, a separate license key is provided for the Kaspersky Embedded Systems Security application that can be used to activate the integration with Kaspersky Managed Detection and Response; this is called the MDR key.

If you did not activate the Kaspersky Embedded Systems Security application during installation (by adding a key to the installation package or by performing initial configuration), you need to activate the installed application in one of the following ways:

- Remotely, using Kaspersky Security Center:
  - Using the Add key task.
  - By distributing a license key stored on the Administration Server to the client devices.
- On the command line using administration commands.

You can also add a reserve key to the application. A reserve key becomes active when the license associated with the active key expires or when the active key is deleted. Availability of a reserve key allows you to avoid application functionality limitation when your license expires.

A reserve license key can be added only after adding an active license key.

You can view information about license keys remotely added to the application using Kaspersky Security Center or the command line on a protected device.

You can also use the graphical user interface to activate the application and manage license keys.

Activating the application using Kaspersky Security Center

You can add license keys to the application through Kaspersky Security Center in the following ways:

- Using the Add key task.

  This method allows you to add a license key to a specific device or the devices included in an administration group. When creating a task, it uses the key that is added to the Kaspersky Security Center key store. You can add a license key to the key store in advance or when creating the activation task.

- By distributing a license key stored on Kaspersky Security Center Administration Server to the client devices.

  This method lets you automatically add a key to the client devices that are already connected to Kaspersky Security Center, and to new client devices. To use this method, first add the key to the Kaspersky Security Center key store.

You can use the Kaspersky Security Center Web Console or Kaspersky Security Center Administration Console to create tasks for adding a key to the application, adding a key to the key store, and distributing the key to the client devices.

Adding keys using the Kaspersky Security Center Web Console.

To add a key to Kaspersky Security Center key storage using the Web Console:

1. In the Web Console main window, select **Operations** → **Kaspersky Licenses**.
2. Click the **Add** button.

3. In the window that opens, select how to add the key to the repository:

   - **Enter the activation code** to add a key using an activation code.
   - **Add a key file** to add a key using a key file.

4. Depending on the key adding method you selected at the previous step, do one of the following:

   - Enter the activation code and click **Submit**.
   - Click the **Select key file** button and in the window that opens, select a file with the .key extension.

5. Click the **Close** button.

The added key will appear in the list of keys.

To add a key to the application via the Web Console using an Add Key task:

1. In the main window of the Web Console, select **Assets (Devices)** → **Tasks**.
   The list of tasks opens.
2. Click the **Add** button.
   The Task Wizard starts.
3. Configure the task settings:
   a. In the **Application** drop-down list, select the application name: Kaspersky Embedded Systems Security.
   b. In the **Task type** drop-down list, select **Add Key**.
   c. In the **Task name** field, enter a brief description, such as `Activation of Kaspersky Embedded Systems Security`.
   d. In the **Devices to which the task will be assigned** section, select the task scope. Click **Next**.
4. Select devices according to the selected task scope option. Click **Next**.
   The **Kaspersky Security Center key storage** window opens.
5. If you have previously added a key to Kaspersky Security Center key storage, select the key from in the list and click **Next**.
6. If the required key cannot be found in the key storage, click the **Add key** button.
   a. In the window that opens, select how to add the key to the repository:
      - **Enter the activation code** to add a key using an activation code.
      - **Add a key file** to add a key using a key file.
   b. Depending on the key adding method you selected at the previous step, do one of the following:
      - Enter the activation code and click **Submit**.
      - Click the **Select key file** button and in the window that opens, select a file with the .key extension.

c. Read the information about the key and click **Close**.

d. The added key will appear in the list of keys. Select it from the list and click **Next**.

7. Read the information about the license and click **Next**.

8. Finish the wizard.

   A new task will be displayed in the list of tasks.

9. Select the check box next to the task. Click the **Start** button.

In the properties of the *Add key* task, you can add a *reserve key* to the application. A reserve key becomes active when the license associated with the active key expires or when the active key is deleted. Availability of a reserve key allows you to avoid application functionality limitation when your license expires.

If you are adding a reserve key but no active key has been added to the application yet, the task ends with an error.

To add a key using the Web Console by distributing a key stored on the Administration Server to the devices:

1. In the Web Console main window, select **Operations** → **Kaspersky Licenses**.

2. Open the key properties using the link with the name of the application for that the key is intended to.

3. On the **General** tab, select the **Automatically distribute a license key to managed devices** check box.

4. Click the **Save** button.

   The license key is automatically distributed to the appropriate client devices.

A new license key is added to a device only if the application has not yet been activated on the device or if the license expires in less than 14 days.

During the automatic distribution of a key as an active or a reserve key, the licensing limit on the number of devices (set in the key properties) is taken into account. If the licensing limit is reached, distribution of this key to the devices stops automatically. You can view the number of devices to which the key has been added and other information in the key properties on the **Devices** tab.

Special considerations for the activation process in Kaspersky Security Center Cloud Console

A trial version is provided for the Kaspersky Security Center Cloud Console. The *trial version* is a special version of Kaspersky Security Center Cloud Console designed to familiarize a user with the features of Cloud Console. In this version, you can perform actions in a workspace for a period of 30 days. All managed applications, including Kaspersky Embedded Systems Security, are automatically activated under Kaspersky Security Center Cloud Console trial license. However, you cannot activate Kaspersky Embedded Systems Security using its own trial license when the trial license for the Cloud Console expires. For more details about Cloud Console, please refer to Kaspersky Security Center Cloud Console documentation.

The trial version of Kaspersky Security Center Cloud Console does not allow you to subsequently switch to a

commercial version. Any trial workspace will be automatically deleted with all its contents after the 30-day period expires.

License key management on the command line

To manage license keys on a device, you can use license key management commands.

To add an active license key to the application, run the following command:

```
kess-control [-L] --add-active-key <path to key file/activation code>
```

where:

- `<path to the key file>` is the path to the key file. If the key file is located in the current directory, it is sufficient to specify only the file name.
- `<activation code>` is the activation code.

To add a reserve license key to the application, run the following command:

```
kess-control [-L] --add-reserve-key <path to key file/activation code>
```

If an active key has not yet been added to the application on the device, the command fails.

To remove an active key, run the following command:

```
kess-control [-L] --remove-active-key [--mdr]
```

To remove a reserve key, run the following command:

```
kess-control [-L] --remove-reserve-key [--mdr]
```

Viewing information about used license keys

You can view information about the license keys being used by Kaspersky Embedded Systems Security in the following ways:

- In Kaspersky Security Center, in the properties of the Add key task.

  In the properties of the *Add key task*, you can find information about the key that this task adds to the application.

- In Kaspersky Security Center, in the properties of the relevant Kaspersky application installed on the client device.

  In the properties of the Kaspersky Embedded Systems Security application on the protected device, you can find information about the active and reserve keys added to the application on this device. You can view the properties of the application using the Web Console as well as the Administration Console.

- In Kaspersky Security Center, in the license key usage report.

  You can view the license key usage report using the Web Console (**Monitoring & reporting → Reports**), as well as the Administration Console (**Reports** tab). To view the report, you need to select the "Report on usage of license keys" template in the list of reports and start generating the report.

- In Kaspersky Security Center, in the Kaspersky Security Center license key store.

  You can open the key store using the Web Console (**Operations → Kaspersky licenses**) or using the Administration Console (**Kaspersky licenses** folder). The store displays information about all keys added to Kaspersky Security Center Administration Server.

- On a device with Kaspersky Embedded Systems Security installed. You can view information about the license used by the Kaspersky Embedded Systems Security application on the command line.

You can also use notifications about Kaspersky Security Center events to get information about the used license keys and the licenses associated with them. The application sends information about expired licenses and license violations to the Kaspersky Security Center Administration Server.

If you use Kaspersky Security Center to manage the application, by default, information about license keys being added and removed and about license term expiration is recorded in the operating system log.

Viewing information about license keys on a device using the Web Console

To view information about license keys added to the device in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices) → Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to view information and click the device name.

4. This opens a managed device properties window; in that window, go to the **Applications** tab.

5. In the list of applications installed on the device, click the name of the Kaspersky Embedded Systems Security application.

6. This opens the application properties window; in that window, select the **General** tab and go to the **Licenses** section.

   This section displays information about license keys added to the application and the licenses associated with these keys.

   - **License key status** is the status of the key: active or reserve.
   - **Application name** is the name of the license associated with the key and information about this license.
   - **License key** is the license key, a unique alphanumeric sequence.
   - **License type** can be trial, commercial, or subscription.

- **Activation date** is the date when this key was added.
- **Expiration date** is the date when your right to use the application activated by adding this key expires.

Viewing information about license keys on a device using the Administration Console

To view information about the license keys added to the device in Kaspersky Security Center Administration Console:

1. In the Kaspersky Security Center Administration Console tree, in the **Managed devices** folder, select the administration group containing the required device.
2. In the workspace, select the **Devices** tab.
3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.
4. In the displayed window with the properties of the managed device, select the **Applications** section.

   The right part of the window displays a list of Kaspersky applications installed on the device.
5. Select Kaspersky Embedded Systems Security and double-click it to open the application properties window. Alternatively, you can use the **Properties** button in the lower part of the window.
6. This opens a window; in that window, go to the **License keys** section.

   This section contains information about the active and reserve license keys:

   - **Serial number** – unique alphanumeric sequence.
   - **Status** – The status of the license key, for example active or reserve.
   - **Type**: type of license (commercial or trial).
   - **License validity period** — Number of days during which you can use the application activated by adding this key.
   - **License limit** — Number of devices on which you can use the key.
   - **Activation date** (this field is only available for the active key): date when the active key was added.
   - **License expiration date** (this field is only available for the active key): date when the application can no longer be used with the current active key.

Viewing information about the license and the key in the command line

In the command line, using the `-L --query` command, you can view information about the active and reserve license keys added to the application, and about the license under which the application has been activated.

To view information about the license keys and license on the device, run the following command:

```
kess-control -L --query [--json]
```

where `--json`: output data in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

As a result of the command execution, the following information will be displayed in the console:

- Information about the active main license key, if an active key has been added:
  - Date and time when the license for using the application expires.
  - Number of days before the end of the license term.
  - Information about the limitation of protection functions.
  - Information about the limitation of the function for updating application databases.
  - Information about the status of the license key.
  - The type of license associated with the key.
  - Licensing limitation of the key (the number of licensing units).
  - Name of the application that the key is intended to activate.
  - Active license key (unique alphanumeric sequence).
  - Activation date.
- Information about the reserve main license key, if a reserve key has been added.
  - Date and time when the license for using the application expires.
  - Information about the limitation of protection functions.
  - Information about the limitation of the function for updating application databases.
  - Information about the status of the license key.
  - The type of license associated with the key.
  - Licensing limitation of the key (the number of licensing units).
  - Name of the application that the key is intended to activate.
  - Reserve license key (unique alphanumeric sequence).
  - Date and time when the license associated with the active key expires, in UTC.

- Information about the active key for additional MDR functionality if an active MDR key has been added:
  - Expiration date and time of the license that activates additional functionality.
  - Number of days before the end of the license term.
  - Information about the limitation of the function for updating application databases.
  - Information about the status of the license key.
  - The type of license associated with the key.
  - Licensing limitation of the key (the number of licensing units).
  - Name of the application that the key is intended to activate.
  - Active license key (unique alphanumeric sequence).
  - Activation date.
- Information about the reserve key for additional MDR functionality if a reserve MDR key has been added:
  - Expiration date and time of the license that activates additional functionality.
  - Information about the limitation of the function for updating application databases.
  - Information about the status of the license key.
  - The type of license associated with the key.
  - Licensing limitation of the key (the number of licensing units).
  - Name of the application that the key is intended to activate.
  - Reserve license key (unique alphanumeric sequence).
  - Date and time when the license associated with the active key expires, in UTC.

You can also get information about the license under which the application is being used when viewing information about the operation of the application using the `kess-control --app-info` command.

Data provision

This section describes the information that Kaspersky Embedded Systems Security may store on the device and automatically send to Kaspersky during its operation.

Kaspersky protects any information thus received in accordance with law and the applicable rules of Kaspersky. Data is transmitted over encrypted channels.

For more detailed information about the processing, storage, and destruction of information obtained during the use of the application and transmitted to Kaspersky, please read the End User License Agreement, the KSN Statement, and refer to the Privacy Policy on the Kaspersky website. The license.<language ID> and ksn_license.<language ID> files containing the End User License Agreement and Kaspersky Security Network Statement are included in the application distribution package.

Data provided when using an activation code

If Kaspersky Embedded Systems Security is activated using an activation code, in order to verify if the application is legally used and to obtain statistical information on the distribution and use of the application, you agree to provide the following information to Kaspersky in automatic mode:

- Type, version, and localization of the installed application
- Versions of installed application updates
- Device ID and application installation ID on the device
- Activation code that was used to activate the application
- ID of the current license
- Application license key creation date and time
- date and time on the user's device;
- Application license term expiration date and time
- Type, version, and bit size of the operating system

Data provided when downloading updates from Kaspersky update servers

If you use Kaspersky update servers to download updates, in order to increase efficiency of the update procedure and to obtain statistical information on distribution and use of the application, you agree to automatically provide to Kaspersky the following information:

- Application ID derived from the license
- Full version of the application
- Application license ID
- Type of application license used
- Application installation ID (PCID)
- ID of the application update start
- Web address being processed

Data sent to Kaspersky Security Center

During operation, Kaspersky Embedded Systems Security saves and submits to Kaspersky Security Center the following information, which may contain personal and confidential data:

- Information about the databases used by the application:
  - List of the database categories required by the application
  - Date and time when the databases were released and loaded into the application
  - Date when the downloaded application database updates were released
  - Time of the last application database update
  - Number of records in the currently used application databases
- Application license information:
  - License serial number and type
  - License validity period in days
  - Number of devices covered by the license
  - Start and end dates of license term
  - License key status

    - Date and time of the last successful synchronization with activation servers if the application was activated using an activation code
    - Identifier of the application for which the license is intended
    - Functionality available under the license
    - Name of the organization for which the license is provided
    - Additional information if the application is used under subscription (subscription flag, subscription expiration date and the number of days available for renewing the subscription, subscription provider web address, current subscription status and the reason for this status), date and time when the application was activated on the device
    - Expiration date and time of the application license on the device
- Information about the application updates:
    - List of updates to be installed or removed
    - Update release date and the sign of the *Critical* status
    - Name, version, and short description of the update
    - Link to the detailed description of the update
    - Identifier and text of the End User License Agreement and the Privacy Policy for the application updates
    - Identifier and text of Kaspersky Security Network Statement for the application updates
    - Indicator showing if the update can be removed
    - Versions of the application policy and management plug-in
    - Web address for downloading the application management plug-in
    - Names, version, and installation dates of the installed application updates
    - Error code and description if the update installation or removal completed with an error
    - Sign and reason for the device or application restart necessity because of the application update
- User agreement or disagreement with the terms and conditions of Kaspersky Security Network Statement, End User License Agreement and Privacy Policy
- List of tags assigned to the device
- List of device statuses and reasons they are assigned.
- The overall status of the application and the status of all its components; information about policy compliance, real-time protection status of the device, application stability status, information about the application stopping.
- Date and time of the last device scan; number of scanned objects; number of detected malicious objects; number of blocked, deleted and disinfected objects; number of objects that cannot be disinfected; number of scan errors; number of detected network attacks

- Data on the currently applied values of the application settings
- Current status and execution results of group tasks and local tasks, and settings of the tasks.
- Information about external devices connected to the client device (ID, name, type, manufacturer, description, serial number, VID/PID)
- Information about backup copies of files in the Backup storage (name, path, size and type of the object, description of the object, name of the detected threat, version of the application database which is used to detect the threat, date and time when the object was moved to the Backup storage), actions on the objects in the Backup storage (removed, restored), and the files by administrator request.
- Information about the operation of each application component and about the execution of each task represented as events:
  - Date and time of event
  - Name and type of event
  - Event severity level
  - Name of the task or the application component running when the event occurred
  - Information about the application that triggered the event: application name, path to the file on the disk, process identifier, setting values (if the application launch or settings modification event is triggered)
  - User ID
  - Name of the initiator (task scheduler, application, Kaspersky Security Center, or a user) whose actions triggered the event
  - Name and identifier of the user who initiated access to the file
  - Object or action processing result (description, type, name, threat level and accuracy, file name and type of operation on the device, application decision on the operation)
  - Information about the object (object name and type, path to the object on the disk, object version, size, information about the performed action, event trigger description, description of the reason for not processing and skipping the object)
  - Device information (manufacturer name, device name, path, device type, bus type, identifier, VID/PID, system device flag, name of the device access rule schedule)
  - Information about blocking and unblocking the device; information about blocked connections (name, description, device name, protocol, remote address and port, local address and port, packet rules, actions)
  - Information about requested web address
  - Information about detected objects
  - Type, method, and ID of the detection
  - Information about the performed action

    - Information about the application databases (date when the downloaded database updates are released, information on the database usage, database usage errors, information on canceling the installed database updates)
    - Information about encryption detection (ransomware name; name of the device where encryption was detected; information about blocking and unblocking the device)
    - Application settings and network settings
    - Information about the triggered Application Control rule (name and type) and the result of applying the rule
    - Information about active and blocked connections (name, description, and type)
    - Information about blocking and unblocking access to untrusted devices
    - Information about the use of KSN (KSN connection status, KSN infrastructure, identifier of the KSN Statement in extended mode, acceptance of the KSN Statement in extended mode, identifier of the KSN Statement, acceptance of the KSN Statement)
    - Information about certificates (domain name, subject name, issuer name, expiration date, certificate status, certificate type, date certificate was added, issue date, serial number, SHA256 thumbprint)
    - Scan task statistics: number of scanned objects; number of threats found; number of infected objects; number of probably infected objects; number of disinfected objects; number of objects added to Backup; number of deleted objects; number of not disinfected objects; number of scan errors; number of password-protected objects; number of skipped objects
    - Information about threat development chains: name of the online list of threat development chains, ID of the threat development chain
- Information about operation of the system integrity scan task (name, type, path) and information about the system baseline
- Information about network activity, packet rules, and network attacks
- User role information:
    - Name and identifier of the user who initiated changing the user role
    - User role
    - Name of the user who has been assigned or revoked the role
- Information about executable files of applications detected on the client device (name, path, type, and hash of the file; list of categories to which the application belongs; KL category to which the application belongs; trust group to which the application belongs; time of the first file launch; name and version of the application; name of the application vendor; information about the certificate used to sign the application: serial number, thumbprint, issuer, subject, release date, expiration date, and public key).

Data provided when following links in the application interface

When clicking the links in Kaspersky Embedded Systems Security interface, you agree to automatically provide the following information to Kaspersky:

- Full version of the application
- Application locale
- Application ID (PID)
- Link name

Data provided when using Kaspersky Security Network

The set of data that Kaspersky Security Network provides to Kaspersky depends on the Kaspersky Security Network mode.

The ksn_license.<language ID> file with the text of the Kaspersky Security Network Statement is included in the application distribution kit.

Data provision in Basic KSN mode

If you are using Kaspersky Security Network in Basic KSN mode (the **Use of Kaspersky Security Network (KSN)** toggle switch is enabled and the **Use Extended KSN mode** check box is cleared in the Web Console, the **Basic KSN mode** option is selected in the Administration Console), by accepting the terms and conditions of the Kaspersky Security Network Statement, you agree to automatically provide the following information to Kaspersky:

- ID of a regional activation center;
- Software activation date and time;
- license expiration date;
- license identifier;
- status of the license used by the Software;
- configuration identifier;
- checksum of the Software activation code;
- full version of the Software;

- unique device ID;
- Software ID;
- checksum of the Software key file;
- ID of the information model used to provide the Software license;
- ID of the certificate used to sign the Software license ticket header;
- creation date and time of the Software license key status confirmation;
- checksum of the Software license key status confirmation;
- version of the Software license key status confirmation;
- Software activation code version;
- active license key status confirmation ID;
- configuration identifier;
- result of the Software action;
- error code;
- checksum type for the object being processed;
- checksum of the object being processed;
- Software component ID;
- version of list of revoked Software service's decisions;
- ID of the triggered record in the Software's anti-virus databases;
- timestamp of the triggered record in the Software's anti-virus databases;
- type of the triggered Software anti-virus databases record;
- name of the detected malware or legitimate software that can be used to damage the user's device or data;
- format of the data in the request to Rightholder's infrastructure;
- accessed address of the web service (URL, IP);
- port number;
- web address of the source of the web service request (referer);
- type of the installed Software.

Data provision in Extended KSN mode

If you are using Kaspersky Security Network in Extended KSN mode (the **Use of Kaspersky Security Network (KSN)** toggle switch is enabled and the **Use Extended KSN mode** check box is selected in the Web Console, the **Extended KSN mode** option is selected in the Administration Console), by accepting the terms and conditions of the Kaspersky Security Network Statement, you agree to automatically provide the following information to Kaspersky in addition to the information listed above:

- performed commands IDs;
- data received date and time;
- ID of the operation being performed on the third-party software;
- third-party software update ID;
- heuristically derived name of the email message sender;
- email message ID;
- event counter;
- event time;
- fragment content of the object being processed;
- unique ID of the activity log for the object being processed;
- date and time when the certificate expires;
- date and time when the certificate was issued;
- CPU architecture;
- unique event ID;
- event date and time;
- total disk space and amount currently used;
- text of the error message;
- name and version of the OS;
- number of update-apply cycles for anti-virus databases;
- date and time when the anti-virus databases were last updated and applied;
- Software categorization base ID;
- Software database record ID;
- Software database record version;

- release date and time of the Software's databases;
- timestamp of the Software databases;
- binary mask of options of the DNS query;
- type of DNS query;
- CPU model;
- CPU brand;
- checksum of the user device name (MD5, SHA2-256, SHA1);
- local port that was attacked;
- date and time on the user's device;
- device ID;
- system log messages containing information about the error;
- OS build number;
- OS update number;
- OS edition;
- extended information about the edition of the OS;
- OS ID;
- OS Service Pack version;
- date and time of the OS launch;
- ID of the account under which the controlled process was started;
- IP address;
- network interfaces list of the computer;
- http request method;
- location where code was injected in process;
- zone index where the endpoint IP address belongs;
- unique User ID in the Rightholder's systems;
- license type;
- ID of the key from the keystore used for encryption;
- computer name on the network (domain name);

- protocol used to exchange data with KSN;
- ID of the KSN service accessed by the Software;
- encryption characteristics of data package that is being sent to KSN;
- ID of data package that is being sent to KSN;
- external IP address;
- local IP address;
- MAC address of the network attack source;
- direction of a network connection;
- source of processed network traffic (local or remote host);
- name of network protocol used in the detected network attack;
- type of user account under which the potentially malicious object was started;
- file attributes of an object being processed;
- fragment order in the object being processed;
- OS boot sectors;
- data of the internal log, generated by the anti-virus Software module for an object being processed;
- result of certificate verification;
- certificate issuer name;
- public key of the certificate;
- calculation algorithm of public key of the certificate;
- certificate serial number;
- date and time of signing the object;
- certificate owner name and settings;
- digital certificate thumbprint of the scanned object and hashing algorithm;
- date and time of the last modification of the object being processed;
- date and time of creating an object being processed;
- detect characteristics;
- object distinguished name;
- file of the email message being processed;

- objects or its parts being processed;
- attributes of executable file being processed;
- date and time of creating an executable file being processed;
- description of an object being processed as defined in the object properties;
- entropy of the file being processed;
- format of the object being processed;
- application image size;
- trust indicator of the processed object according to KSN;
- result of status check in KSN of an object being processed;
- number of software runs since the last time the file checksum was sent;
- date and time of linking the executable file;
- checksum (MD5) of the object being processed;
- name of the object being processed;
- names of the packers that packed the object being processed;
- Software name;
- checksum (MD5) of the mask that blocked the web service;
- checksum (SHA256) of the object being processed;
- information on who signed the file being processed;
- size of the object being processed;
- flag indicating an application which runs automatically at startup;
- object type code;
- Software vendor name;
- Software verdict on the object being processed;
- version of the object being processed;
- time of the first launch of the object being processed;
- source of the decision made for the object being processed;
- source of the object being processed;
- parent application name;

- integrity level for the object being processed;
- IP address from which the file that matches the process was downloaded;
- segments of random access memory;
- Software module ID;
- result of the module integrity check;
- address for the Software module loading;
- detected file operations with the object being processed;
- result of action with the object being processed;
- file operation code;
- path to the object being processed;
- directory code;
- process system ID (PID);
- command line;
- full names of files that were accessed by the object being processed;
- IP address that was accessed by the object being processed;
- access rights for the object being processed;
- information about file signature check results;
- source file path;
- vulnerability ID;
- vulnerability danger class;
- detect location within the web traffic being processed;
- file of the web page being processed;
- data packages of the web traffic being processed;
- window ID for the application being processed;
- bitness of the OS;
- full path to parent process file used to launch the process;
- parent process system ID (PID);
- command line arguments for the process;

- date and time of System Watcher start;
- version of the Software's component;
- data about the license for identifying a group of users of the company that purchased the license by the comment in the license properties;
- Software update ID;
- installation date and time for the Software;
- Software installation type;
- number of devices/accounts covered by the Software license;
- nested error occurred during the application operation;
- notification type, that triggered the statistic sending;
- full version of the Software;
- information about Software updates;
- Software installation ID (PCID);
- Software component name;
- number of the string in the script where the error has occurred;
- Software health status after update;
- name of the created/modified OS service;
- logon session key;
- encryption algorithm for the logon session key;
- memory stack of the Software process failure;
- user story type;
- attribute data;
- web address from which the file that matches the process was downloaded;
- DNS server response;
- IP address of the DNS server;
- name of the module in which the failure probably occurred;
- date and time when statistics stopped being received;
- error type;

- user actions with the interface element in the application window;
- object time in the buffer;
- IP address of the attacker;
- debug detection indicator;
- attribute of an object being processed, that allowed to recall the false positive decision on the object;
- ID of the task in which detection was performed;
- number of KSN connections taken from the cache;
- number of requests for which a response was found in the local request database;
- number of unsuccessful KSN connections;
- number of unsuccessful KSN transactions;
- temporal distribution of canceled requests to KSN;
- temporal distribution of unsuccessful KSN connections;
- temporal distribution of unsuccessful KSN transactions;
- temporal distribution of successful KSN connections;
- temporal distribution of successful KSN transactions;
- temporal distribution of successful requests to KSN;
- temporal distribution of requests to KSN that timed out;
- number of new KSN connections;
- number of unsuccessful requests to KSN caused by routing errors;
- number of unsuccessful requests caused by KSN being disabled in the Software settings;
- number of unsuccessful requests to KSN caused by network problems;
- number of successful KSN connections;
- number of successful KSN transactions;
- total number of requests to KSN;
- statistics message type;
- confidence of detecting access to the phishing web service;
- phishing attack target;
- weight of the detected access to the phishing web service;

- protocol ID;
- ID of the operation being performed by the Software;
- date and time when statistics started being received;
- probability of sending statistics by System Watcher;
- code of the event that took longer than the standard time to process by System Watcher;
- database processing time of the event that took longer than the standard time to process by System Watcher;
- processing delay time of the event about OS action in the behavioral analysis subsystem;
- number of delayed OS action events of the current type;
- maximum allowed time for processing an event by System Watcher;
- processing delay time of the event about OS action in the proactive defense subsystem;
- number of processed OS action events;
- number of processed synchronous OS action events;
- total delay of all OS action events of the current type;
- processing delay time of the event about OS action in the persistent event storage subsystem;
- processing time of the event that took longer than the standard time to process by System Watcher;
- total number of events that took longer than the standard time to process by System Watcher;
- total delay of all OS action events;
- number of waiting synchronous OS action events;
- date and time of detecting software by System Watcher;
- number of the detected software in the System Watcher context;
- reason of detecting software by System Watcher;
- date and time of received event of an action in the OS;
- version of the statistics being sent;
- technical specifications of the applied detection technologies;
- version of a certain compiler;
- properties and checksums of the parts of the execution file;
- emulator version;
- emulation depth;

- type of executable file scan task that sends statistics;
- storage time for object being processed;
- algorithm for calculating the digital certificate thumbprint;
- component name;
- timestamp of the update component (updated version);
- timestamp of the component (local version);
- number of failed update installations for the updater component;
- code of the error category;
- number of update installation error for the updater component;
- timestamp of the root index of updates being downloaded;
- timestamp of the root index of available updates;
- error code of the update task;
- value of the update task TARGET filter;
- update task type;
- version of the updater component;
- binary mask of the parameters for the object being processed;
- checksum of the user name;
- user account security identifier (SID);
- DNS address of the web service being accessed;
- host source;
- header of the http request being processed;
- accessed IPv4 address of the web service;
- accessed IPv6 address of the web service;
- indicator showing that the message is a part of a bundle of messages belonging to one access to the web service;
- web address being processed;
- information about the client that uses a network protocol (user agent);
- security zone ID from the NTFS stream;

- Software ID derived from the license;
- Software license ID;
- serial number of the Software license key;
- Software localization;
- identifier of the partner organization via which the Software license order was placed;
- Software rebranding ID;
- operating status of the Software component;
- ID of the licensed Software;
- flag indicating whether participation in KSN is enabled;
- delay of sending the statistics.

Application management concept

To manage Kaspersky Embedded Systems Security, you can use:

- Kaspersky Security Center
- The command line
- The graphical user interface

The set of actions that you can perform using the Kaspersky Embedded Systems Security graphical user interface is limited.

This section describes the specifics of managing the application via Kaspersky Security Center and the command line, and also describes the main methods of working in the Kaspersky Security Center administration consoles and in the command line.

Managing the application using Kaspersky Security Center

Kaspersky Security Center allows you to remotely and centrally manage the operation of Kaspersky Embedded Systems Security on client devices. You can remotely install and uninstall, start, and stop Kaspersky Embedded Systems Security; configure settings for the application, as well as for the individual components and tasks of the application; and start and stop tasks on the managed devices.

You can use the following Kaspersky Security Center administration consoles to manage Kaspersky Embedded Systems Security via Kaspersky Security Center:

- Kaspersky Security Center Administration Console (hereinafter also referred to as Administration Console). This is a Microsoft Management Console (MMC) snap-in that is installed on the administrator's workstation and provides a user interface for the Administration Server and Network Agent administrative services.

  The interface for managing Kaspersky Embedded Systems Security via the Kaspersky Security Center Administration Console is provided by the administration MMC plug-in (hereinafter also referred to as the "MMC plug-in").

- Kaspersky Security Center Web Console (hereinafter also referred to as Web Console). This is a web interface for managing a protection system based on Kaspersky applications. You can work in Kaspersky Security Center Web Console using a browser on any device that has access to the Administration Server.

  The interface for managing Kaspersky Embedded Systems Security via the Kaspersky Security Center Web Console is provided by the administration web plug-in (hereinafter also simply referred to as web plug-in).

- Kaspersky Security Center Cloud Console. This is a cloud-based administration console within the cloud version of the Kaspersky Security Center application, also known as the Kaspersky Security Center Cloud Console. Interface of the Cloud console is similar to Kaspersky Security Center Web Console interface. The interface for managing Kaspersky Embedded Systems Security via the Kaspersky Security Center Cloud Console is also provided by the web plug-in.

The MMC plug-in and web plug-in allow you to create policies and tasks in Kaspersky Security Center for managing the operation of Kaspersky Embedded Systems Security:

- A *policy* is a set of settings that is applied on all devices in an administration group. Policies allow you to apply identical application settings to all client devices within an administration group.

  The Kaspersky Embedded Systems Security policy defines the general settings for the operation of Kaspersky Embedded Systems Security and the settings for the operation of individual functional components of the application on devices where the policy is applied.

- *Tasks* for Kaspersky Embedded Systems Security created in Kaspersky Security Center run on the protected devices and implement Kaspersky Embedded Systems Security functions such as on-demand scan, application activation, and updates to the databases and modules of the application.

  In Kaspersky Security Center, you can create tasks to be performed on an individual device (local tasks), tasks for all devices in the administration group (group tasks), or tasks for a random selection of devices (tasks for sets of devices).

Regardless of the Kaspersky Security Center administration console that you use, you must assign the devices on which Kaspersky Embedded Systems Security is installed to administration groups in order to manage Kaspersky Embedded Systems Security on these devices using Kaspersky Security Center. You can create administration groups in Kaspersky Security Center before Kaspersky Embedded Systems Security installation and configure rules to automatically move the devices to administration groups. You can also manually move the devices to the administration groups after installing the Kaspersky Embedded Systems Security (for details, refer to Kaspersky Security Center documentation).

About Kaspersky Embedded Systems Security management plug-ins

The following management plug-ins are required for managing Kaspersky Embedded Systems Security using Kaspersky Security Center:

- Kaspersky Embedded Systems Security administration web plug-in (hereinafter also referred to as the *web plug-in*) facilitates interaction between Kaspersky Embedded Systems Security and Kaspersky Security Center using Kaspersky Security Center Web Console and Kaspersky Security Center Cloud Console.

  The web plug-in must be installed on the device that has Kaspersky Security Center Web Console installed. Management of Kaspersky Embedded Systems Security using the web plug-in is available to all administrators who have access to the Kaspersky Security Center Web Console in a browser.

- The Kaspersky Embedded Systems Security administration MMC plug-in (hereinafter also referred to as the *MMC plug-in*) facilitates interaction between Kaspersky Embedded Systems Security and Kaspersky Security Center using the Administration Console.

  The MMC plug-in must be installed on the device where the Kaspersky Security Center Administration Console is installed.

The Kaspersky Embedded Systems Security management plug-ins let you manage Kaspersky Embedded Systems Security using policies and tasks.

For more details about management plug-ins, refer to Kaspersky Security Center documentation.

Kaspersky Security Center policies

A *policy* is a set of Kaspersky Embedded Systems Security settings that are applied to all client devices included in the administration group.

Multiple policies with different values of the settings can be configured for a single application. However, there can be only one active policy at a time for an application within an administration group. When you create a new policy, all other policies within an administration group become inactive. You can change the policy status later.

Policies have a hierarchy, similarly to administration groups. By default, a child policy inherits settings from the parent policy. A *child policy* is a policy of a nested hierarchy level, that is, a policy for nested administration groups and secondary Administration Servers. You can enable inheritance of the settings from the parent policy.

You can locally modify the values of the settings specified by the policy for individual devices within the administration group, if modification of these settings is not prohibited by the policy.

Each policy setting has a "lock" attribute that indicates whether child policy settings and local application settings can be modified. The "lock" status of a setting within policy properties determines whether or not an application setting on a client device can be edited:

- When a setting is "locked" (), you cannot edit its value locally or in the policies of the nested hierarchy level. The setting value specified by the policy is used for all client devices within the administration group and nested groups.

- When a setting is "unlocked" (), you can edit its value locally or in the policies of the nested hierarchy level. If setting values are specified locally or in policy properties of a nested hierarchy level for client devices within an administration group, the setting value specified in the policy properties is not applied.

In the web plug-in and in the MMC plug-in, the number of parameters with "locks" is different. The web plug-in includes "locks" that are not present in the MMC plug-in.

For some lists, you can enable the merging of list items. To merge list items, you need to do the following:

- In the properties of the child policy, on the **General** tab, enable the **Inherit settings from parent policy** setting.

- In the properties of the parent policy, click the padlock () icon to "lock" it.

- In the properties of the parent policy, enable the merging of settings by selecting the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy of Kaspersky Security Center are displayed in child policies, and items can be added to the list of child policies. If the check box is cleared, the list items are not merged when inheriting policy settings.

You cannot delete or modify inherited list items in a child policy. List items that have been merged when inherited can be deleted or modified only in the parent policy. List items can be added, modified, or removed at lower levels. Thus, you can configure a list in the parent policy, which will be applied to all child policies, and you can also add to the list in the child policy.

A *policy profile* may contain settings that differ from the "base" policy settings and apply to client devices when the configured conditions (activation rules) are met. Using policy profiles allows you to flexibly configure application settings for different devices. You can create and configure profiles in the **Policy profiles** section of the policy properties.

Profile settings that are locked with a "padlock" override policy settings. That is, if the profile setting locked with a "padlock" is different from the policy setting, the application applies the setting from the profile. However, lists of settings are merged, supplementing each other. That is, if the settings in the list from the profile are missing from the "basic" policy, they are added to the resulting list of settings.

Some lists are not merged. If this happens, the settings from the profile override the settings of the "base" policy:

- Exclusions by process in the File Threat Protection and Behavior Detection components
- Protection scopes in the File Threat Protection and Anti-Cryptor components
- Monitoring scopes in the System Integrity Monitoring component
- List of rules (in the **Application Control rules** window) in the Application Control component
- Process memory exclusions in application settings
- Trusted domains in network settings
- Trusted root certificates in network settings
- Monitored ports in network settings

After the policy is applied for the first time, the application settings change in accordance with the policy settings.

If the application is not running when the policy is deleted, after application is started, this policy continues to be applied on the device and the application continues to operate with the settings specified by this policy.

For more information about policies and policy profiles, as well as the settings inheritance mechanism, see the Kaspersky Security Center Help.

Tasks for Kaspersky Embedded Systems Security created in Kaspersky Security Center

You can create the following types of tasks in Kaspersky Security Center for Kaspersky Embedded Systems Security:

- local tasks to run on individual devices;
- group tasks to run on devices within an administration group;
- Tasks for sets of devices to run on multiple devices, regardless of their inclusion in administration groups

The tasks for the sets of devices are performed only on the devices that are specified in the task settings. If new devices are added to the device selection for which the task is created, this task is not applied to the new devices. To apply the task to these computers, you must create a new task or edit the settings of the existing task.

You can create any number of group tasks, tasks for a sets of devices, or local tasks.

The tasks are executed only if Kaspersky Embedded Systems Security is running on the devices.

General information about tasks created in Kaspersky Security Center is provided in Kaspersky Security Center documentation.

The following tasks are provided for managing Kaspersky Embedded Systems Security in Kaspersky Security Center:

- *Malware Scan*. During the task execution, the application scans the device areas that are specified in the task settings for viruses and other malware.
- *Critical Areas Scan*. During the task execution, the application scans boot sectors, startup objects, process memory, and kernel memory.
- *Inventory*. During the task execution, the application receives information about all executable files stored on the devices.
- *System Integrity Check*. During the task execution, the application determines changes of each object by comparing the current state of the monitored object to its original state, which was previously established as a baseline.
- *Add Key*. During the task execution, the application adds a key, including a reserve one, to activate the application.
- *Update*. During the task execution, the application updates the databases in accordance with the configured update settings.
- *Rollback*. During the task execution, the application rolls back the last database update.

Logging in and out of the Web Console and Cloud Console

Kaspersky Security Center Web Console

To log in to the Web Console, you need to know the web address and the port number of the Administration Server specified during the Web Console installation (port 8080 is used by default). JavaScript must also be enabled in your browser.

To log in to Web Console:

1. In your browser, go to `<Administration Server web address>:<port>`.

   The login page is displayed.

2. Enter the user name and password for your account.

   It is recommended to make sure that the password complexity and anti-bruteforce mechanisms ensure that the password cannot be guessed within 6 months.

3. Click **Log in**.

   If the Administration Server is not responding, or if you enter incorrect credentials, an error message is displayed.

After logging in, a dashboard is displayed with the last language and theme used.

For more details about the Web Console interface, see the Kaspersky Security Center Help.

To log out of Web Console:

select **<Account name>** → **Exit** in the lower left corner of the screen.

The Web Console is closed, and the login page is displayed.

Kaspersky Security Center Cloud Console

For the Kaspersky Security Center Cloud Console, use a web token to log in to your account on the Cloud Console portal.

For detailed information about Kaspersky Security Center Cloud Console, refer to Kaspersky Security Center Cloud Console documentation.

Managing policies in the Web Console

You can perform the following actions with the policies in the Web Console:

- Create a policy.
- Edit policy settings. You can configure policy settings on the **Application settings** tab of the policy properties window.

  If the user account which is used to access the Administration Server does not have permissions to edit the settings of certain functional scopes, the settings of these functional scopes are not available for editing.

- Export and import policy settings.

  Exporting or importing policy settings from Kaspersky Embedded Systems Security 3.4 for Linux to Kaspersky Embedded Systems Security 4.0 for Linux is not supported.

- Copy and move a policy.
- Delete a policy.
- Change a policy status.
- Create policy profiles.

For general information about working with policies, refer to the Kaspersky Security Center Help system.

Creating a policy in the Web Console

To create a policy in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Policies and policy profiles**.

   A list of policies and policy profiles opens.

2. Select the administration group containing the devices to which the policy should be applied. To do so, click the link in the **Current path** field located above the list of policies and policy profiles, and select the administration group in the window that opens.

3. Click the **Add** button.

   The Policy Wizard starts.

4. In the displayed window, select an application name from the list.

   Proceed to the next step of the wizard.

5. Decide whether you want to use Kaspersky Security Network. Carefully read the Kaspersky Security Network Statement and do one of the following:

   - If you agree with all the terms and conditions of the Statement and want the application to use Kaspersky Security Network, select **I confirm that I have fully read, understand, and accept the terms and conditions of Kaspersky Security Network Statement**.

   - If you do not want to use Kaspersky Security Network, select **I do not accept the terms and conditions of the Kaspersky Security Network Statement** and confirm your decision in the window that opens.

   Refusal to use Kaspersky Security Network does not interrupt the policy creation process. At any time, you can enable or disable use of Kaspersky Security Network or change the KSN mode for managed devices in the policy settings.

   Proceed to the next step of the wizard.

6. The **General** tab of the new policy settings window opens. Specify a name for the new policy.

   You can also configure the following policy settings:

   - Policy status:

     - **Active**. The policy that is currently applied to the device. If this option is selected, this policy becomes active on the device upon the next device synchronization with the Administration Server. This option is selected by default.

     - **Inactive**. The policy that is not currently applied to the device. If this option is selected, the policy becomes inactive but remains in the **Policies** folder. You can activate the inactive policy later.

- **Out-of-office**. If this option is selected, the policy becomes active when the device leaves the organization network.

  In the Network Agent policy that is active on the device, in the **Network → Connection profiles** section, the **Enable out-of-office mode when Administration Server is not available** setting must be enabled.

- Policy settings inheritance:

  - **Inherit settings from parent policy**. If this toggle switch is enabled, the policy settings are inherited from the upper-level group policy. Settings that are locked with a "padlock" (🔒) in the parent policy are not editable. The toggle switch is switched on by default.

  - **Force inheritance of settings in child policies**. If this toggle switch is enabled, the policy settings are propagated to child policies. In the properties of the child policy, the **Inherit settings from parent policy** toggle button will be automatically switched on and cannot be switched off. The settings of the child policy are inherited from the parent policy, except for the settings for which the "padlock" (🔓) is open. The toggle button is switched off by default.

For general information about the policy settings, refer to Kaspersky Security Center Help section.

7. If you want to configure other policy settings, go to the **Application settings** tab and make the necessary changes.

   You can also change the policy settings later.

8. Click the **Save** button.

The created policy will be displayed in the list of policies.

For general information about managing policies, please refer to the Kaspersky Security Center Help.

Changing policy settings in the Web Console

To edit policy settings in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Policies and policy profiles**.

   The list of policies opens.

2. Select the administration group containing the devices to which the policy is applied. To do so, click the link in the **Current path** field in the upper part of the window and select the administration group in the window that opens.

   The list displays the policies configured for the selected administration group.

3. Click the name of the required policy in the list.

   The policy properties window opens.

4. Modify the policy settings on the **Application settings** tab.

5. Click the **Save** button to save the changes made.

The policy is saved with the updated settings.

Policy settings in the Web Console

You can configure policy settings on the **Application settings** tab of the policy properties window.

Policy settings

Section	Subsections
Host Threat Protection	**File Threat Protection**. In this subsection, you can manage File Threat Protection. **BadUSB Attack Prevention**. In this subsection, you can manage BadUSB Attack Prevention settings. **Removable Drives Scan**. In this subsection, you can configure the settings for scanning removable drives when they are connected to a protected device. **Behavior Detection**. In this subsection, you can manage settings for application behavior detection. **Anti-Cryptor**. In this subsection, you can configure the settings for protection against remote malicious encryption.
Communication & Web Protection	**Web Threat Protection**. In this subsection, you can configure Web Threat Protection settings. **Network Threat Protection**. In this subsection, you can configure Network Threat Protection settings. **Firewall**. In this subsection, you can configure firewall management settings.
Security Controls	**Device Control**. In this subsection, you can configure Device Control settings. **App Control**. In this subsection, you can configure Application Control settings. **System Integrity Monitoring**. In this subsection, you can configure System Integrity Monitoring settings.
General Settings	**Detection Settings**. In this subsection, you can manage the following settings: • Detection of legitimate applications that threat intruders can use to compromise devices or data. • File operations interception mode. **Kaspersky Security Network**. In this subsection, you can configure the use of Kaspersky Security Network by Kaspersky Embedded Systems Security. **User Interaction**. In this subsection, you can manage the following settings: • Application startup settings. • Task management permissions for users. **Performance**. In this subsection, you can manage the following settings: • Canceling scheduled tasks on a device running on battery power. • Exclude process memory from scans. • Limit on the use of memory and processor resources for scan tasks. • Limit on the use of resident memory by the application. • Application stability monitoring. **Network Settings**. In this subsection, you can configure the following: • Proxy server settings if the users of the client devices use a proxy server to connect to the internet. • Network traffic scan settings. **Container Scan**. In this subsection, you can configure the use of the namespace mechanism. **Technical Support**. In this subsection, you can manage the trace and dump settings. **Reports and Storage**. In this subsection, you can manage the settings of Backup and Quarantine. **KSC Integration**. In this subsection, you can configure the following: • Using Kaspersky Security Center as a proxy server. • Settings for the transfer of data to the Kaspersky Security Center storage. **OS interaction settings**. In this subsection, you can manage additional telemetry settings. **Notification settings**. In this subsection, you can manage event notification settings.
Built-in Agents Configuration	**Managed Detection and Response**. In this subsection, you can enable or disable the integration with Kaspersky Managed Detection and Response.
Trusted zone	In this section, you can manage the following settings: • Exclusions that are applied by the File Threat Protection component. • Exclusions that are applied by the Web Threat Protection component. • Exclusions that are applied by the Network Threat Protection component. • Exclusions from network traffic scanning. • Excluding mount points from file operation interception. • Trusted devices for the Device Control component.

Managing policies in the Administration Console

You can perform the following actions with the policies in the Kaspersky Security Center Administration Console:

- Create a policy.
- Edit policy settings. You can manage policy settings in the policy properties window.

  If the user account which is used to access the Administration Server does not have permissions to edit the settings of certain functional scopes, the settings of these functional scopes are not available for editing.

- Export and import policy settings.
- Delete a policy.
- Change a policy status.
- Create policy profiles.

For general information about managing policies, please refer to the Kaspersky Security Center Help.

Creating a policy using the Administration Console

To create a policy in the Administration Console:

1. In the Administration Console tree, in the **Managed devices** folder, select the administration group containing the devices to which the policy should be applied.

   You can view the list of devices that are part of an administration group on the **Devices** tab of the folder with the name of this administration group.

2. In the workspace, select the **Policies** tab.
3. Click the **New policy** button to start the New policy wizard.

   You can also start the Wizard by clicking the **Create** → **Policy** item in the context menu in the list of policies.

4. At the first step of the wizard, select **Kaspersky Embedded Systems Security 4.0 for Linux** from the list.

   Proceed to the next step of the wizard.

5. Enter a name for the new policy.
6. To use the settings from the previous version of Kaspersky Embedded Systems Security policy in the policy being created, select the **Use policy settings for the earlier application version check box**.

   Proceed to the next step of the wizard.

7. Decide whether you want to use Kaspersky Security Network. Carefully read the Kaspersky Security Network Statement and do one of the following:

   - If you agree with all the terms and conditions of the Statement and want the application to use Kaspersky Security Network, select **I confirm that I have fully read, understand, and accept the terms and conditions of Kaspersky Security Network Statement**.
   - If you do not want to use Kaspersky Security Network, select **I do not accept the terms and conditions of the Kaspersky Security Network Statement** and confirm your decision in the window that opens.

   Refusal to use Kaspersky Security Network does not interrupt the policy creation process. At any time, you can enable or disable use of Kaspersky Security Network or change the KSN mode for managed devices in the policy settings.

   Proceed to the next step of the wizard.

8. If necessary, configure the general settings for File Threat Protection.

   Proceed to the next step of the wizard.

9. If necessary, edit the File Threat Protection settings that have been configured by default.

   Proceed to the next step of the wizard.

10. If necessary, configure the exclusions from File Threat Protection.

    Proceed to the next step of the wizard.

11. If necessary, modify the default actions for infected objects.

    Proceed to the next step of the wizard.

12. Select a policy state:

    - **Active policy**. The policy that is currently applied to the device. If this option is selected, this policy becomes active on the device upon the next device synchronization with the Administration Server. This option is selected by default.
    - **Inactive policy**. If this option is selected, the policy is not applied to the device, but remains in the **Policies** folder. You can activate the inactive policy later.
    - **Out-of-office policy**. If this option is selected, the policy becomes active when the device leaves the organization network.

      In the Network Agent policy that is active on the device, in the **Network → Connection profiles** section, the **Enable out-of-office mode when Administration Server is not available** setting must be enabled.

13. Complete the New Policy Wizard.

The created policy is displayed in the list of policies of the administration group on the **Policies** tab and in the **Policies** folder of the console tree.

You can change the policy settings later. For general information about managing policies, refer to the Kaspersky Security Center Help.

Changing policy settings in the Kaspersky Security Center Administration Console

To edit policy settings in the Administration Console:

1. In the tree of the Kaspersky Security Center Administration Console, in the **Managed devices** folder, open the folder with the name of the administration group that includes the required devices.
2. In the workspace, select the **Policies** tab.
3. In the list of policies, select the required policy and double-click it to open the **Properties: <Policy name>** window.

   You can also open the policy properties window by using the **Properties** item in the policy context menu or by clicking the **Configure policy settings** link located to the right of the list of policies in the section with the policy settings.
4. Edit the policy settings.
5. In the **Properties: <Policy name>** window, click **OK** to save the changes.

Policy settings in the Administration Console

You can configure policy settings in the sections and subsections of the policy properties window. For information about configuring general policy settings and event settings, refer to Kaspersky Security Center Help section.

Policy settings

Section	Subsections
Essential Threat Protection	**File Threat Protection**. In this subsection, you can manage File Threat Protection. **File Threat Protection exclusions**. In this subsection, you can configure exclusions that apply to the File Threat Protection component. **Firewall Management**. In this subsection, you can configure firewall management settings. **Web Threat Protection**. In this subsection, you can configure Web Threat Protection settings. **Network Threat Protection**. In this subsection, you can configure Network Threat Protection settings. **BadUSB Attack Prevention**. In this subsection, you can manage BadUSB Attack Prevention settings.
Advanced Threat Protection	**Kaspersky Security Network**. In this subsection, you can configure the use of Kaspersky Security Network by Kaspersky Embedded Systems Security. **Anti-Cryptor**. In this subsection, you can configure the settings for protection against remote malicious encryption. **Behavior Detection**. In this subsection, you can manage settings for application behavior detection.
Detection and Response	**Managed Detection and Response**. In this subsection, you can enable or disable the integration with Kaspersky Managed Detection and Response.
Security Controls	**Application Control**. In this subsection, you can configure Application Control settings. **System Integrity Monitoring**. In this subsection, you can configure System Integrity Monitoring settings. **Device Control**. In this subsection, you can configure Device Control settings.
Local Tasks	**Task Management**. In this subsection, you can configure permissions for users to manage tasks. **Removable Drives Scan**. In this subsection, you can configure the settings for scanning removable drives when they are connected to a protected device.

Section	Subsections
General settings	**Proxy server settings**. In this subsection, you can configure the proxy server settings if users access the internet from client devices through a proxy server. **Application settings**. In this subsection, you can manage the following settings: • Detection of legitimate applications that threat intruders can use to compromise devices or data. • Notification settings. • Limit on the use of memory and processor resources for scan tasks. • Limit on the number of Custom Scan tasks that a non-privileged user can start on a device at the same time. • Application startup settings. • Trace and dump settings. • Application stability monitoring. • Limit on the use of resident memory by the application. • File operations interception mode. **Container Scan settings**. In this subsection, you can configure the use of the namespace mechanism. **Network settings** In this subsection, you can configure the settings for scanning network traffic. **Global exclusions**. In this subsection, you can configure the following: • Excluding mount points from file operation interception. • Industrial software exclusions to exclude processes, files, and directories associated with a particular type of software. **Excluding process memory**. In this subsection, you can configure exclusion of process memory from scans. **Storage settings**. In this subsection, you can configure the following: • Backup and quarantine. • Settings for the transfer of data to the Kaspersky Security Center storage. **OS interaction settings** In this subsection, you can configure the following: • Select the system event interception mechanism that you want the application to use. • Manage advanced telemetry parameters settings.

Managing tasks in the Web Console

You can perform the following actions with the tasks for Kaspersky Embedded Systems Security in the Web Console:

- Create new tasks.
- Edit task settings.

  If the user account which is used to access the Administration Server does not have permissions to edit the settings of certain functional scopes, the settings of these functional scopes are not available for editing.

- Start, stop, pause, and resume tasks.

  The *Update* task cannot be paused or resumed, it can only be started or stopped.

- Export and import tasks.
- Delete tasks.

In the list of tasks, you can monitor the task execution results: view the task status and the statistics for task performance on the devices. You can also create a selection of events to monitor the task execution (**Monitoring and reports → Event selections**). For details on event selection, refer to Kaspersky Security Center documentation.

Task execution results are also saved locally on the device and in Kaspersky Security Center reports.

For general information about task management, refer to the Kaspersky Security Center Help system.

If the device is managed by a policy, it may not be possible to view and manage tasks created in Kaspersky Security Center using the command line or the graphical interface of the application.

Creating tasks in the Web Console

To create a task for a group or set of devices in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Tasks**.

   The list of tasks opens.

2. Click the **Add** button.

   The Task Wizard starts.

3. In the first step of the Wizard, perform the following actions:

   a. In the **Application** drop-down list, select the application name: Kaspersky Embedded Systems Security.

   b. In the **Task type** drop-down list, select the type of task that you want to create.

   c. In the **Task name** field, enter a name for the new task.

   d. In the **Devices to which the task will be assigned** section, select the method for defining the task scope. The task scope comprises the devices on which the task will be run:

   - Select the **Assign task to an administration group** option if the task is to be run on all devices included in a specific administration group.
   - Select the **Specify device addresses manually, or import addresses from a list** option if the task is to be run on the specified devices.
   - Select the **Assign task to a device selection** option if the task is to be run on devices included in the device selection according to a predefined criterion. For information on how to create a device selection, refer to the Kaspersky Security Center Help system.

   Proceed to the next step of the wizard.

4. Depending on the selected method for defining the task scope, perform one of the following actions:

   - In the administration group tree, select the check boxes next to the required administration groups.
   - In the list of devices, select the check boxes next to the required devices. If the required devices are not listed, you can add them in the following ways:
     - Using the **Add devices** button. You can add devices by name or IP address, add devices from a specified IP range, or select devices from the list of devices detected by the Administration Server when polling the corporate LAN.
     - Using the **Import devices from file** button. For the import, a TXT file with a list of device addresses is used, where each address must be on a separate line.
   - From the list, select the name of the selection containing the required devices.

   Proceed to the next step of the wizard.

5. To configure the task settings immediately after creation, in the last step of the Wizard, select the **Open task properties window after creation** check box. A task is created with the default settings.

6. Finish the wizard.

A new task will be displayed in the list of tasks.

To create a local task in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to create a task and click the device name.

4. This opens a managed device properties window; in that window, go to the **Tasks** tab.

   The list of tasks created for this device is displayed.

5. Click the **Add** button.

   The Task Wizard starts.

6. In the first step of the Wizard, perform the following actions:

   a. In the **Application** drop-down list, select the application name: Kaspersky Embedded Systems Security.

   b. In the **Task type** drop-down list, select the type of task that you want to create.

   c. In the **Task name** field, enter a name for the new task.

7. To configure the task settings immediately after creation, in the last step of the Wizard, select the **Open task properties window after creation** check box. A task is created with the default settings.

8. Finish the wizard.

A new task will be displayed in the list of tasks.

Changing task settings in the Web Console

To edit task settings in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Tasks**.

   The list of tasks opens.

2. Do one of the following:

   - To edit the settings of a task that is run on all devices included in a specific administration group, click the link in the **Current path** field in the upper part of the window and select the administration group in the window that opens.

     The list displays only tasks configured for the selected administration group.

   - To edit the settings of a task that is run on one or multiple devices (a task for a set of devices), click the link in the **Current path** field in the upper part of the window and select the top node with the name of the Administration Server in the window that opens.

     The list displays all tasks created on the Administration Server.

3. In the list of tasks, select the required task and open the task properties window by clicking the link in the task name.

4. Configure the task settings:

   - On the **General** tab, you can edit the name of the task.

   - On the **Application settings** tab, you can configure specific task settings. The availability of configurable settings depends on the type of task.

   - On the **Schedule** tab, you can configure the task run schedule and additional settings for starting and stopping the task.

   The **General**, **Results**, **Settings**, **Schedule**, and **Revision history** tabs of the task properties window are standard for Kaspersky Security Center; for more details, refer to the Kaspersky Security Center Help system.

5. Click the **Save** button to save the changes made.

Starting, stopping, pausing, and resuming tasks in the Web Console

To start, stop, pause, or resume a task in the Web Console:

1. In the main window of Kaspersky Security Center Web Console, select **Assets (Devices)** → **Tasks**.

   The list of tasks opens.

2. Do one of the following:

   - To start or stop a task that is run on all devices included in a specific administration group, click the link in the **Current path** field in the upper part of the window and select the administration group in the window that opens.

     The list displays only the tasks created for the selected administration group.

   - To start or stop a task that is run on one or multiple devices (a task for a set of devices), click the link in the **Current path** field in the upper part of the window and select the top node with the name of the Administration Server in the window that opens.

     The list displays all tasks created on the Administration Server.

3. In the list of tasks, check the box next to the name of the required task and click the action button above the list of tasks.

Managing tasks in the Administration Console

You can perform the following actions with the tasks for Kaspersky Embedded Systems Security in the Administration Console:

- Create new tasks.
- Edit task settings.

  If the user account which is used to access the Administration Server does not have permissions to edit the settings of certain functional scopes, the settings of these functional scopes are not available for editing.

- Start, stop, pause, and resume tasks.

  The *Update* task cannot be paused or resumed, it can only be started or stopped.

- Export and import tasks.
- Delete tasks.

In the list of tasks, you can monitor the task execution results: view the task status and the statistics for task performance on the devices.

Information on the progress and results of task execution can be viewed in the list of events that Kaspersky Embedded Systems Security sends to the Kaspersky Security Center Administration Server (on the **Events** tab in the workspace of the **Administration Server <server name>** node). You can also create a selection of events to monitor the execution of tasks. For details on event selection, refer to Kaspersky Security Center documentation.

Task execution results are also saved locally on the device and in Kaspersky Security Center reports.

For general information about task management, refer to the Kaspersky Security Center Help system.

If the device is managed by a policy, it may not be possible to view and manage tasks created in Kaspersky Security Center using the command line or the graphical interface of the application.

Creating tasks in the Administration Console

To create a task for a group or set of devices in the Administration Console:

1. In the Administration Console, perform one of the following actions:

   - To create a task that will be run on devices included in the selected administration group, select this administration group in the console tree in the **Managed devices** folder, then select the **Tasks** tab in the workspace and click the **New task** button.

     The New task wizard starts for devices of the selected administration group.

   - To create a task that will be performed on one or multiple devices (a task for a set of devices), select the **Tasks** folder in the console tree and click the **New task** button in the workspace.

     The New task wizard starts for the set of devices.

2. At the first step of the wizard, select Kaspersky Embedded Systems Security and a task type.

   Proceed to the next step of the wizard.

3. If you are creating a task for a set of devices, the Wizard prompts you to define the task scope. The task scope comprises the devices on which the task will be run.

   a. Specify the method for defining the task scope: select devices from the list of devices detected by the Administration Server; set device addresses manually; import a list of devices from a file or specify a previously configured selection of devices (for more details, refer to the Kaspersky Security Center Help system).

   b. Depending on the method you have specified for defining the task scope, in the window that opens, perform one of the following actions:

      - In the list of detected devices, specify the devices on which the task will be run. To do so, select the check box in the list to the left of the device name.

      - Click the **Add** or **Add IP range** button and enter the device addresses manually.

      - Click the **Import** button and select the TXT file containing the list of device addresses in the window that opens.

      - Click the **Browse** button and, in the window that opens, specify the name of the selection containing the devices on which the task will be run.

   Proceed to the next step of the wizard.

4. Configure the available task settings by following the instructions in the Wizard.

5. Enter the name of the new task and proceed to the next step in the Wizard.

6. To start the task immediately after the Wizard finishes, in the final step, select the **Run task after the wizard finishes** check box.

7. Finish the wizard.

   A new task will be displayed in the list of tasks.

To create a local task in the Administration Console:

1. In the Administration Console tree, in the **Managed devices** folder, select the administration group containing the necessary device.

2. In the workspace, select the **Devices** tab.

3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.

4. In the displayed window with the properties of the managed device, select the **Tasks** section.

   The list of tasks created for this device is displayed.

5. Click the **Add** button.

   The Task Wizard starts.

6. At the first step of the wizard, select Kaspersky Embedded Systems Security and a task type.

   Proceed to the next step of the wizard.

7. Enter a name for the new task and configure the available task settings following the instructions of the wizard.

8. Finish the wizard.

   A new task will be displayed in the list of tasks.

Changing task settings in the Administration Console

To edit task settings in the Administration Console:

1. In the Administration Console, perform one of the following actions:

   - To edit the settings of a task that is run on devices included in the specified administration group, select this administration group in the console tree, then select the **Tasks** tab in the workspace.

   - To edit the settings of a task that is run on one or multiple devices (a task for a set of devices), select the **Tasks** folder in the console tree.

2. In the list of tasks, select the required task and double-click it to open the **Properties: <Task name>** window.

   You can also open the task properties window using the **Properties** item in the task context menu.

3. Edit the task settings. The availability of configurable settings depends on the type of task.

   The **General**, **Notification**, **Schedule**, and **Revision history** tabs of the task properties window are standard for Kaspersky Security Center; for more details, refer to the Kaspersky Security Center Help system.

4. Click **Apply** or **OK** in the **Properties: <Task name>** window to save the changes made.

Starting, stopping, pausing, and resuming tasks in the Administration Console

To start, stop, pause, or resume a task in the Administration Console:

1. In the Administration Console, perform one of the following actions:

   - To start or stop a task that is run on devices included in the specified administration group, select this administration group in the console tree, then select the **Tasks** tab in the workspace.

     The list of tasks created for the selected administration group opens.

   - To start or stop a task that is run on one or multiple devices (a task for a set of devices), select the **Tasks** folder in the console tree.

     The list of all tasks created on the Administration Server opens.

2. In the list of tasks, select the required task, open the context menu of the task, and select the action that you want to perform.

Managing the application using the command line

Using the command line, you can install, uninstall, start, and stop Kaspersky Embedded Systems Security on the device, and also manage the application locally.

The functional components of the application are supported by *Kaspersky Embedded Systems Security local tasks* that run in the operating system. You can enable or disable functional components of the application on a device by starting or stopping Kaspersky Embedded Systems Security tasks in the command line. One-time device scans are also performed by starting Kaspersky Embedded Systems Security tasks. You can define the settings for functional components on the device and the device scan settings by configuring the Kaspersky Embedded Systems Security *task settings*.

In addition to the task settings, the following settings are provided for configuring the application:

- Encrypted connections scan settings.
- General application settings that define the operation of the application as a whole and the operation of individual functions.

On the command line, Kaspersky Embedded Systems Security can be managed using *Kaspersky Embedded Systems Security management commands*.

Enabling automatic addition of kess-control commands (bash completion)

Kess-control commands can be automatically added for the bash shell.

To enable automatic addition of kess-control commands in the current bash shell session, run the following command:

```
source /opt/kaspersky/kess/shared/bash_completion.sh
```

To enable automatic addition for all new bash shell sessions, run the following command:

```
echo "source /opt/kaspersky/kess/shared/bash_completion.sh" >> ~/.bashrc
```

Task management on the command line

The following application tasks are provided for managing Kaspersky Embedded Systems Security using the command line:

- *File Threat Protection*. This task allows you to enable or disable File Threat Protection in real time and defines the settings for the File Threat Protection component. The task starts automatically when the application starts.
- *Malware Scan*. This task allows you to scan file system objects for malware on demand and defines the settings for the scan. You can use this task to perform a full or custom scan of the device.
- *Critical Areas Scan*. This task allows you to run a critical areas scan of the operating system on demand and defines the settings for the scan.
- *Custom file scan*. This task is designed for configuring and storing settings that are used when scanning the specified files and directories using the `kess-control --scan-file` command. As a result of the command execution, the application creates and starts a temporary file scan task.
- *Removable Drives Scan*. This task allows you to monitor the connection of removable media to the device in real time and defines the settings of the Removable Drives Scan and the scan of its boot sectors for malware.
- *Web Threat Protection*. This task allows you to enable or disable Web Threat Protection and defines the settings for the Web Threat Protection component.
- *Network Threat Protection*. This task allows you to enable or disable Network Threat Protection and defines the settings for the Network Threat Protection component.
- *Anti-Cryptor*. This task allows you to enable or disable the protection of files from remote malicious encryption and defines the settings for the Anti-Cryptor component.
- *Firewall Management*. This task allows you to enable or disable firewall management and defines the network connection control settings on the device.
- *Application Control*. This task allows you to enable or disable Application Control and defines the settings of the Application Control component.
- *Inventory*. The task allows you to obtain information about all the application executable files stored on the device.
- *Device Control*. This task allows you to enable or disable Device Control and defines the settings for the Device Control component. The task starts automatically when Kaspersky Embedded Systems Security starts.
- *Behavior Detection*. This task allows you to monitor malicious activity of applications in the operating system. The task starts automatically when Kaspersky Embedded Systems Security starts.

- *System Integrity Monitoring*. This task allows you to perform real-time monitoring of the actions performed with objects from the monitoring scope specified in the System Integrity Monitoring component settings.

- *System Integrity Check*. This task allows you to check for changes in files and directories that you have included in the monitoring scope, by comparing the current state of the monitored object with a previously recorded state.

- *Licensing*. This task provides the capability to activate an application installed on the device. The task starts automatically when the application starts, and it resides in the device operating memory. The task has no settings; license keys are managed using special management commands. The task cannot be started, stopped, or deleted.

- *Update*. You can use this task to perform scheduled and on-demand application database and module updates and edit update settings.

- *Rollback*. You can use this task to roll back the last update of application databases and modules.

Each application task has a name used on the command line, an ID, and a type (see the table below).

IDs are unique for all tasks, including deleted tasks. The application does not reuse the identifiers of the deleted tasks. The identifier of a new task is the next successive number to the identifier of the latest created task.

Task names are not case-sensitive.

During installation of the application, *predefined tasks* are created. These tasks cannot be deleted. Each predefined task has a name and ID.

Tasks that you create while working with the application are called *user tasks*. When you create the task, you specify the name for it. IDs for user tasks are defined and assigned by the application when the task is created. IDs for user tasks are starting from 100.

During operation, the application creates *temporary scan tasks*. Temporary task names and IDs are assigned by the application. Temporary tasks are automatically deleted when completed.

Application tasks

Task	Task name in command line	Task ID	Task type
File Threat Protection	File_Threat_Protection	1	OAS
Malware Scan	Scan_My_Computer	2	ODS
Malware Scan (user task)	user-defined	starting from 100	ODS
Custom file scan	Scan_File	3	ODS
Critical Areas Scan	Critical_Areas_Scan	4	ODS
Update	Update	6	Update
Update (user task)	user-defined	starting from 100	Update
Rollback	Rollback	7	Rollback
Rollback (user task)	user-defined	starting from 100	Rollback
Licensing	License	9	License
System Integrity Monitoring	System_Integrity_Monitoring	11	OAFIM
System Integrity Monitoring (user task)	user-defined	starting from 100	ODFIM
Firewall Management	Firewall_Management	12	Firewall
Anti-Cryptor	Anti_Cryptor	13	AntiCryptor
Web Threat Protection	Web_Threat_Protection	14	WTP

Task	Task name in command line	Task ID	Task type
Device Control	Device_Control	15	DeviceControl
Removable Drives Scan	Removable_Drives_Scan	16	RDS
Network Threat Protection	Network_Threat_Protection	17	NTP
Behavior Detection	Behavior_Detection	20	BehaviorDetection
Application Control	Application_Control	21	AppControl
Inventory	Inventory_Scan	22	InventoryScan
Inventory (user task)	user-defined	starting from 100	InventoryScan
BadUSB Attack Prevention	Bad_USB	37	BadUSB

You can perform the following actions with tasks:

- Start and stop all predefined and user tasks except the *License* task.
- Suspend and resume *ODS*, *ODFIM*, and *InventoryScan* tasks.
- Create and delete user tasks. You can create the following types of tasks: *ODS*, *Update*, *Rollback*, *ODFIM* and *InventoryScan*.
- Change the settings for all user tasks and all predefined tasks, except for *Rollback* and *License* tasks.
- Configure the task start schedule.

Viewing a list of tasks in the command line

To view the list of application tasks, execute the following command:

```
kess-control --get-task-list [--json]
```

where:

`--json` – output format for the list of application tasks. If a file format is not specified, the output will be an INI file.

The list of Kaspersky Embedded Systems Security tasks will be displayed.

The following information will be displayed for each task:

- `Name`: the task name
- `ID`: the task ID
- `Type`: the task type
- `State`: the current state of the task

If the Kaspersky Security Center policy prohibits users from viewing and editing local tasks, information about the Scan_My_Computer, Critical_Areas_Scan, Inventory_Scan, Update, and Rollback tasks is not available.

Viewing the status of a task in the command line

To view a task state, execute the following command:

```
kess-control --get-task-state <task ID/name> [--json]
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--json` is specified to output the settings in JSON format.

Application tasks can take the following main states:

- `Started`—Task is running.
- `Starting`—Task is being launched.
- `Stopped`—Task has been stopped.
- `Stopping`—Task is stopping.

The *ODS*, *ODFIM*, and *InventoryScan* tasks can also have one of the following states:

- `Pausing` — Task is pausing.
- `Suspended` — Task is suspended.
- `Resuming` — Task is resuming.

Creating a task in the command line

You can create the following types of tasks: *ODS*, *Update*, *Rollback*, *ODFIM*, and *InventoryScan*.

You can create tasks with default settings or with settings specified in a configuration file.

To create a task with default settings, execute the following command:

```
kess-control --create-task <task name> --type <task type>
```

where:

- `<task name>` is the name that you specify for the new task.
- `<task type>` is the designation of the type of the task you are creating.

To create a task with the settings specified in the configuration file, execute the following command:

```
kess-control --create-task <task name> --type <task type> --file <path to
configuration file> [--json]
```

where:

- `<task name>` is the name that you specify for the new task.
- `<task type>` is the identifier for the type of the task you are creating.
- `<path to file>` is the full path to the configuration file with the settings that will be used for creating the task.
- `--json` is specified to import the settings from the configuration file in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

Starting, stopping, pausing, and resuming tasks in the command line

You can start and stop predefined and user tasks, except for tasks of the *License* type.

You can suspend and resume tasks of *ODS*, *ODFIM*, and *InventoryScan* types.

To start a task, execute the following command:

```
kess-control --start-task <task ID/name> [-W] [--progress]
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `[-W]` enables the output of current events related to this task.
- `[--progress]` enables the display of task progress.

Example:
Start the task with ID 1 and enable the display of current events associated with the task:

```
kess-control --start-task 1 -W
```

If an error occurs when starting a task and the task does not start, then after the application is restarted, an attempt is made to start the task again.

To stop a task, execute the following command:

```
kess-control --stop-task <task ID/name> [-W]
```

where:

- `< task ID/name >` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `[-W]` enables the output of current events related to this task.

To suspend a task, execute the following command:

```
kess-control --suspend-task < task ID/name >
```

To resume a task, execute the following command:

```
kess-control --resume-task < task ID/name >
```

Deleting a task in the command line

You can delete only user tasks. Predefined tasks cannot be deleted.

To delete a task, execute the following command:

```
kess-control --delete-task < task ID/name >
```

where `< task ID/name >` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

Displaying task settings in the command line

You can display the current values of settings for all user tasks and all predefined tasks, except for *Rollback* and *License* tasks (these tasks have no settings).

You can output the current values of task settings to the console or to a configuration file that you can use to change task settings.

To output the current values of task settings to the console, execute the following command:

```
kess-control --get-settings < task ID/name > [--json]
```

where:

- `< task ID/name >` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

To output the current values of task settings to a configuration file, execute the following command:

```
kess-control --get-settings <task ID/name> --file <path to configuration file> [--
json]
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--file <configuration file path>` is the path to the configuration file where the task settings will be written. If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, the configuration file will not be created.
- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

Editing task settings in the command line

You can edit the settings for all user tasks and all predefined tasks, except for *Rollback* and *License* tasks.

On the command line, you can edit the settings of tasks using the `kess-control --set-settings` command:

- You can edit all task settings using the configuration file that contains the task settings. You can get the configuration file using the command for displaying task settings.
- You can edit individual task settings using command line options in the `<setting name>=<setting value>` format. You can get the current values of task settings using the command for displaying task settings.
- You can restore the task settings to their default values.

You can add or remove scan scopes and exclusion scopes using a configuration file that contains task settings or command line options. Configuring scan scopes and exclusion scopes is available for tasks with the *OAS*, *ODS*, *OAFIM*, *ODFIM*, and *AntiCryptor* types.

In order to optimize the operation of scan tasks, it is recommended to add the path with snapshots mounted by the system in the read-only mode to the exclusions for the systems with the btrfs file system and enabled active snapshots. For example, for the systems based on SUSE/OpenSUSE, you can add the following exclusion for the path: `/.snapshots/*/snapshot/`.

For some tasks, separate management commands are also provided that allow you to edit task settings.

Editing task settings using a configuration file

To edit values of task settings using a configuration file:

1. Output the task settings to the configuration file using the command `kess-control --get-settings`.

2. Open the configuration file and edit the values of the necessary settings.

   For tasks of the *OAS*, *ODS*, *OAFIM*, *ODFIM*, and *AntiCryptor* types, you can add or remove scan scopes and exclusion scopes.

   If you want to add a scan scope, add a `[ScanScope.item_#]` section with the following settings to the file:

   - `AreaDesc` is a description of the scan scope, which contains additional information about this scope.
   - `UseScanArea` enables scanning of the specified scope.
   - `Path` is a path to the directory with the objects to be scanned. You can specify a path to a local directory or enable scanning of remote directories mounted on a client device.
   - `AreaMask.item_#` is a limitation of the scan scope. You can specify a mask for the name of the files to be scanned. Scanning is enabled by default for all objects in the scan scope. You can specify multiple `AreaMask.item_#` items.

   If you want to add an exclusion scope, add an `[ExcludedFromScanScope.item_#]` section with the following settings to the file:

   - `AreaDesc` – a description of the exclusion scope, which contains additional information about the exclusion scope.
   - `UseScanArea` enables exclusion of the specified scope.
   - `Path` is a path to the directory with the objects to be excluded. You can specify a path to a local directory or exclude remote directories mounted on a client device. Possible values for the setting depend on the type of task.
   - `AreaMask.item_#` is a limitation of the exclusion scope. You can specify a mask for the name of the files that you want to exclude from the scan scope. By default, all objects in the scope are excluded.

     Example:
     ```
     [ExcludedFromScanScope.item_0000]
     AreaDesc=
     UseScanArea=Yes
     Path=/tmp/notchecked
     AreaMask.item_0000=*
     ```

   You can specify multiple `[ScanScope.item_#]` and `[ExcludedFromScanScope.item_#]` sections. The application processes the scopes by index in ascending order.

3. Save the configuration file.

4. Execute the command:

   `kess-control --set-settings <task ID/name> --file <path to configuration file> [--json]`

where:

- `< task ID/name >` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--file < configuration file path >` is the full path to the configuration file from which the task settings will be imported.
- Specify the `--json` option if you are importing settings from a JSON configuration file. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

All values of task settings defined in the file will be imported into the application.

If you change the allowlist, or prohibit launch of all applications or applications that affect the operation of Kaspersky Embedded Systems Security in the Application Control task settings, run the `--set-settings` command with the `--accept` option.

Editing task settings using the command line options

Using the `kess-control --set-settings` command options, you can edit individual values of task settings, as well as add or remove scan scopes and exclusion scopes for tasks of the *OAS*, *ODS*, *OAFIM*, *ODFIM*, and *AntiCryptor* types.

Configuring individual task settings

To edit individual values of task settings using command line options, execute the following command:

```
kess-control --set-settings < task ID/name > < setting name >=< setting value > [< setting name >=< setting value >]
```

where:

- `< task ID/name >` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `< setting name >=< setting value >` is the name and value of one of the task settings. You can get the current values of task settings using the command for displaying task settings.

The values of the specified task settings will be changed.

If you change the allowlist, or prohibit launch of all applications or applications that affect the operation of Kaspersky Embedded Systems Security in the Application Control task settings, run the `--set-settings` command with the `--accept` option.

Adding and removing a scan scope

To add a scan scope using command line options, execute the following command:

```
kess-control --set-settings <task ID/name> --add-path <path>
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--add-path <path>` adds a path to a directory with objects to be scanned.

A new `[ScanScope.item_#]` section will be added to the task settings. The application scans the objects in the directory specified by the `Path` setting. The remaining settings of the scan scope take default values.

If the task settings already contain an `[ScanScope.item_#]` section with the specified value for the `Path` setting, a duplicate section is not added.

If the `UseScanArea` setting is set to `No` its value will change to `Yes` after this command is executed and the objects located in this directory will be scanned.

Example:
Adding a scan scope for a task with ID=100:

```
kess-control --set-settings 100 ScanScope.item_0001.UseScanArea=Yes ScanScope.item_0001.Path=/home
```

The following scan scope settings will be added to the task:

```
[ScanScope.item_0001]

AreaDesc=

UseScanArea=Yes

Path=/home

AreaMask.item_0000=*
```

To delete a scan scope using command line options, execute the following command:

```
kess-control --set-settings <task ID/name> --del-path <path>
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--del-path <path>` removes a path to a directory with objects to be scanned.

The `[ScanScope.item_#]` section that contains the specified path will be deleted from the task settings. The application will not scan the objects in the specified directory.

Adding and removing an exclusion scope

To add an exclusion scope using command line options, execute the following command:

```
kess-control --set-settings <task ID/name> --add-exclusion <path>
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--add-exclusion <path>` adds a path to a directory with objects that you want to exclude from the scan.

A new `[ExcludedFromScanScope.item_#]` section will be added to the task settings. The application will exclude objects in the directory specified by the `Path` setting from scans. The remaining settings of the exclusion scope take default values.

If the task settings already contain an `[ExcludedFromScanScope.item_#]` section with the specified value for the `Path` setting, a duplicate section is not added.

If the `UseScanArea` setting is set to `No` its value will change to `Yes` after this command is executed and the objects located in this directory will be excluded from scans.

To delete an exclusion scope using command line options, execute the following command:

```
kess-control --set-settings <task ID/name> --del-exclusion <path>
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--del-exclusion <path>` removes a path to a directory with objects to be excluded from scanning.

The `[ExcludedFromScanScope.item_#]` section that contains the specified path will be deleted from the task settings. The application will not exclude the objects in the specified directory from the scan.

Restoring default task settings in the command line

You can restore the default settings for all user tasks and all predefined tasks, except for tasks of the *Rollback* and *License* types (these tasks have no settings).

To reset task settings to their default values, execute the following command:

```
kess-control --set-settings <task ID/name> --set-to-default
```

where `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

The application changes the setting values to their defaults.

Configuring task schedule in the command line

You can configure the schedule for running the following types of tasks: *ODS*, *Update*, *Rollback*, *ODFIM*, and *InventoryScan*.

You can output the current values of the settings for the task run schedule to the console or to a configuration file.

To output the current settings for the task run schedule to the console, execute the following command:

```
kess-control --get-schedule <task ID/name> [--json]
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

To output the current settings for the task run schedule to a configuration file, execute the following command:

```
kess-control --get-schedule <task ID/name> --file <path to configuration file> [--
json]
```

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `--file <configuration file path>` is the path to the configuration file where the task schedule settings will be written. If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, the configuration file will not be created.
- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

Examples:
Save the update task settings to a file named update_schedule.ini and save the created file in the current directory:

```
kess-control --get-schedule 6 --file update_schedule.ini
```

Display the update task schedule in the console:

```
kess-control --get-schedule 6
```

You can edit the settings for the task run schedule in the following ways:

- Import the settings from a configuration file that contains all schedule settings.
- Using the command line, specify individual settings of the task schedule in the `<setting name>=<setting value>` format.

To edit the values of the settings for task run schedule using a configuration file, perform the following actions:

1. Output the task settings to the configuration file using the `kess-control --get-schedule` command.
2. Edit the values of the necessary settings in the file and save the changes.
3. Execute the command:

   `kess-control --set-schedule <task ID/name> --file <path to configuration file> [--json]`

   where:

   - `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
   - `--file <configuration file path>` is the full path to the configuration file from which the task schedule settings will be imported.
   - Specify the `--json` option if you are importing settings from a JSON configuration file. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

   All values of the settings for the task run schedule defined in the file will be imported into the application.

Example:
Import the schedule settings from the configuration file named /home/test/on_demand_schedule.ini into the task with ID=2:

```
kess-control --set-schedule 2 --file /home/test/on_demand_schedule.ini
```

To edit the individual values of the settings for the task run schedule using the command line, execute the following command:

`kess-control --set-schedule <task ID/name> <setting name>=<setting value> [<setting name>=<setting value>]`

where:

- `<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.
- `<setting name>=<setting value>` is the name and value of one of the task schedule settings.

The values of the specified settings for the task run schedule are modified.

Examples:

To schedule the task to start every ten hours, specify the following settings:

```
RuleType=Hourly

RunMissedStartRules=No

StartTime=2025/May/30 23:05:00;10

RandomInterval=0
```

To schedule the task to start every ten minutes, specify the following settings:

```
RuleType=Minutely

RunMissedStartRules=No

StartTime=23:10:00;10

RandomInterval=0
```

To schedule the task to start on the 15th of every month, specify the following settings:

```
RuleType=Monthly

RunMissedStartRules=No

StartTime=23:25:00;15

RandomInterval=0
```

To schedule the task to start on every Tuesday, specify the following settings:

```
RuleType=Weekly

StartTime=18:01:30;Tue

RandomInterval=99

RunMissedStartRules=No
```

To schedule the task to start every 11 days, specify the following settings:

```
RuleType=Daily

RunMissedStartRules=No

StartTime=23:15:00;11

RandomInterval=0
```

Managing general application settings on the command line

General application settings define the operation of the application as a whole and the operation of individual functions.

You can manage general application settings using special management commands:

- Output the current values of general application settings to the console or to a configuration file.
- Edit general application settings using a configuration file containing all general settings, or using command line options in < `setting name` >=< `setting value` >. format.

Using general settings, you can:

- Configure the use of Kaspersky Security Network and the light version of anti-malware databases in the application.
- Configure the use of a proxy server in the application.
- Select the file operation interception mode (block or do not block files during a scan).
- Configure exclusions from the mount points scan (global exclusions).
- Configure exclusions from the process memory scan.
- Enable or disable the detection of legitimate applications that intruders can use to compromise devices or data.
- Configure the use of event logs.
- Configure a limit on CPU resource usage by scan tasks (of the ODS type).
- Limit the number of user scan tasks that a non-privileged user can start simultaneously.

Displaying general application settings

You can output the current values of general application settings to the console or to a configuration file that you can use to edit application settings.

To output the current values of general application settings to the console, execute the following command:

```
kess-control --get-app-settings [--json]
```

where `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

To output the current values of general application settings to a configuration file, execute the following command:

```
kess-control --get-app-settings --file <path to configuration file> [--json]
```

where:

- `--file <configuration file path>` is the path to the configuration file where the application general settings will be written. If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, the configuration file will not be created.

- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

Example:
Display the general application settings to a file named kess_config.ini.Save the created file in the current directory:

```
kess-control --get-app-settings --file kess_config.ini
```

Editing general application settings

On the command line, you can edit the general application settings using the command `kess-control --set-app-settings`:

- You can edit all general settings using the configuration file that contains the general application settings. You can get the configuration file using the command for displaying general settings.

- You can edit individual settings using command line options in the format `<setting name>=<setting value>`. You can get the current values of general application settings using the command for displaying general settings.

To edit values of general application settings using a configuration file:

1. Output the general application settings to a configuration file.

2. Edit the values of the necessary parameters in the file and save the changes.

3. Execute the command:

   `kess-control --set-app-settings --file <path to configuration file> [--json]`

   where:

   - `--file <path to configuration file>` is the full path to the configuration file with the general application settings. If you delete any parameter in the configuration file, the default value for that parameter will be displayed after the command is executed.

   - Specify the `--json` option if you are importing settings from a JSON configuration file. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

   All the values of the general settings defined in the file will be imported into the application.

To edit values of general application settings using command line options, execute the following command:

`kess-control --set-app-settings <setting name>=<setting value> [<setting name>=<setting value>]`

where < `setting name` >=< `setting value` > is the name and value of one of the general application settings.

The values of the specified general settings will be changed.

Examples:

Import general settings into the application from the configuration file /home/test/kess_config.ini:

```
kess-control --set-app-settings --file /home/test/kess_config.ini
```

Set the detail level for the trace file to low:

```
kess-control --set-app-settings TraceLevel=NotDetailed
```

Add a mount point that you want to exclude from interception of file operations:

```
kess-control --set-app-settings ExcludedMountPoint.item_0000="/data"
```

Using filters to limit results of queries

A filter allows you to limit the query results when executing application management commands.

Filter conditions are specified using one or more *logical expressions*, which are combined using the logical operator `and`. Filter conditions must be enclosed in quotation marks:

"< `field` > < `comparison operator` > '< `value` >'"

"< `field` > < `comparison operator` > '< `value` >' and < `field` > < `comparison operator` > '< `value` >'"

where:

- < `field` > is the name of the database field.
- < `comparison operator` > is one of the following comparison operators:
  - > is "greater than"
  - < is "less than"
  - `like` matches the specified value When specifying a value, you can use % masks: for example, the logical expression "FileName like '%etc%'" sets the limitation "contains the text "etc" in the FileName field"

    The `like` comparison operation works only with fields that are stored in the database as text. The `like` operation is not suitable for fields with dates, times, numbers, or other data types other than text.

  - == is "equal to"
  - != is "not equal to"
  - >= is "greater than or equal to"
  - <= is "less than or equal to"

- < value > is the value of the field. The value must be enclosed in single quotation marks (').

  You can specify a date value as UNIX™ time (the number of seconds that have elapsed since 00:00:00 (UTC), January 1, 1970) or in YYYY-MM-DD hh:mm:ss format. The user specifies the date and time in the user's local time zone, and the application displays them in the same time zone.

You can use a filter in the following application management commands:

- Output information about certain current application events (Viewing events on the command line, Viewing events in Kaspersky Security Center):

  ```
  kess-control -W --query "< filter conditions >"
  ```

- Output information about certain application events (Viewing events on the command line, Viewing events in Kaspersky Security Center) in the event log:

  ```
  kess-control -E --query "< filter conditions >"
  ```

- Display information about certain objects in the Backup:

  ```
  kess-control -B --query "< filter conditions >"
  ```

- Delete certain objects from the Backup:

  ```
  kess-control -B --mass-remove --query "< filter conditions >"
  ```

  Examples:

  Get information about events that contain the text "etc" in the FileName field:

  ```
  kess-control -E --query "FileName like '%etc%'"
  ```

  Display information about events with the ThreatDetected type:

  ```
  kess-control -E --query "EventType == 'ThreatDetected'"
  ```

  Display information about events with the ThreatDetected type, created by tasks of the ODS type:

  ```
  kess-control -E --query "EventType == 'ThreatDetected' and TaskType == 'ODS'"
  ```

  Get information about the events generated after the date specified in the UNIX time stamp system (the number of seconds that have elapsed since 00:00:00 (UTC), 1 January 1970):

  ```
  kess-control -E --query "Date > '1583425000'"
  ```

  Get information about the events generated after the date specified in YYYY-MM-DD hh:mm:ss format:

  ```
  kess-control -E --query "Date > '2022-12-22 18:52:45'"
  ```

  Get information about files in the Backup storage that have the High severity level:

  ```
  kess-control -B --query "DangerLevel == 'High'"
  ```

Exporting and importing application settings

If Kaspersky Embedded Systems Security is managed via Kaspersky Security Center, importing settings is not supported.

Kaspersky Embedded Systems Security allows you to export and import all application settings for troubleshooting, verifying settings, or simplifying the application's configuration on other user devices. When exporting settings, all application settings (including encrypted connections scan settings, general application settings, and task settings) are saved in a configuration file. You can use this configuration file to import settings into the application.

The application must be launched when settings are imported or exported. After the settings are imported, the application must be restarted.

When importing or exporting settings from an older application version, new settings are set to default values. Importing settings to an older application version is not supported.

To export the application settings, execute the following command:

```
kess-control --export-settings --file <path to configuration file> [--json]
```

where:

- `--file <configuration file path>` is the full path to the configuration file where the application settings will be saved.
- `--json` is specified to export the settings to the configuration file in JSON format. If the `--json` options is not specified, the settings will be exported to an INI file.

To import the application settings from the file, execute the following command:

```
kess-control --import-settings --file <path to configuration file> [--json]
```

where:

- `--file <configuration file path>` is the full path to the configuration file from which the settings will be imported into the application.
- `--json` is specified to import the settings from the configuration file in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

When you import application settings from a file, the `UseKSN` and `CloudMode` settings are set to `No`. To start or resume the use of Kaspersky Security Network, set the value of the `UseKSN` setting to `Basic` or `Extended`. To enable cloud mode, you must set the `CloudMode` setting to `Yes`. Cloud mode is available if use of KSN is enabled.

After application settings are imported, internal task IDs may change. It is recommended to use task names to manage tasks.

Managing user roles using the command line

Access to Kaspersky Embedded Systems Security functions via the command line is provided to users in accordance with their roles. A *role* is a set of rights and privileges for managing the application.

The four groups of system users are created in the operating system: *kessadmin*, *kessuser*, *kessaudit*, and *nokess*. When you assign an application role to a system user, the user is added to the corresponding group (see the *Roles* table below). When you revoke a role from a user, this user is removed from the corresponding group.

If no application role is assigned to a system user, that user belongs to a separate group of *users without rights*.

Thus, the roles correspond to the four groups of operating system users:

- kessadmin – the Administrator role
- kessuser – the User role
- kessaudit – the Auditor role
- nokess is assigned to a user if no other roles are assigned. In this case, the user belongs to a separate group of *users without privileges*

User roles

Role name	Role in application	OS user group	Permissions
Administrator	admin	kessadmin	Managing application settings and task settings in the graphical interface of the application and on the command line without using the `sudo` command. Managing application licensing. Assigning roles to users. Revoking user roles (the administrator has no right to revoke the admin role from self). View and manage users' Storages.
User	user	kessuser	Manage only user file scan tasks. Start and stop Update tasks. View reports for the tasks created by this user. View specific events that are common for all application users.
Auditor	audit	kessaudit	Viewing application settings View application status. View all tasks, their settings, and start schedules. View all events. View all objects in Backup.
—	—	nokess	No role is assigned in the application, no permissions.

Viewing a list of users and roles

To view a list of users and their roles, execute the following command:

```
kess-control [-U] --get-user-list
```

Assigning a role to a user

To assign a role to a specific user, execute the following command:

```
kess-control [-U] --grant-role <role> <user>
```

Example:
To assign the audit role to the user test15:

```
kess-control --grant-role audit test15
```

Revoking a user role

To revoke a role from a specific user, execute the following command:

```
kess-control [-U] --revoke-role <role> <user>
```

Example:
To revoke the audit role from the user test15:

```
kess-control --revoke-role audit test15
```

Starting and stopping the application

After installing the Kaspersky Embedded Systems Security to a device, the application is started automatically. By default, the application then starts automatically when the operating system is booted (at the default level of execution for each operating system).

By default, when Kaspersky Embedded Systems Security is started, the following functional components of the application are started automatically:

- File Threat Protection.
- Device Control.
- Behavior Detection.
- Web Threat Protection—only if one of the supported browsers is installed in the operating system and local management of Web Threat Protection settings is allowed on the device (a policy is not applied or the "lock" is not set in the policy properties).
- Network Threat Protection—only if the Network Threat Protection settings on the device are defined through a policy. Network Threat Protection is enabled in the policy properties by default. If locally configured settings are applied on the device, Network Threat Protection is disabled by default.

When the application is started, service tasks are automatically started on the device to ensure the operation of additional application functions: the application activation function and the Backup function.

By default, the application also starts user tasks configured on the command line, for which the "after application startup" run mode (`PS` run mode) is configured.

If you stop the application, all tasks running on the device will be interrupted. Interrupted user tasks are not resumed automatically after the application is restarted.

Starting and stopping the application using the Web Console

To start or stop the application remotely:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. In the list, select the device on which you want to start or stop the application, and click the link with the device name to open the device properties window.

3. Select the **Applications** tab.

4. Select the Kaspersky Embedded Systems Security check box.

5. Do one of the following:

   - To start the application, click the **Start** button.
   - To stop the application, click the **Stop** button.

You can monitor the application operation status by using the **Protection status** widget in the **Monitoring and reports / Dashboard** window.

Starting and stopping the application using the Administration Console

To start or stop the application on a client device:

1. In the Administration Console tree, in the **Managed devices** folder, select the administration group containing the necessary device.
2. In the workspace, select the **Devices** tab.
3. In the list of managed devices, select the device for which you want to start or stop the application. In the device context menu, select **Properties**.
4. In the **Properties: <Device name>** window, select the **Applications** section.

   The right part of the window displays a list of Kaspersky applications installed on the device.
5. Select the Kaspersky Embedded Systems Security application.
6. Do one of the following:

   - To run the application, click the button to the right of the list of Kaspersky applications or select **Start** in the application context menu.
   - To stop the application, click the button to the right of the list of Kaspersky applications or select **Stop** in the application context menu.

Starting and stopping the application using the command line

To run the application, the root account must be the owner of the following directories and only the owner must have write access to them: /var, /var/opt, /var/opt/kaspersky, /var/log/kaspersky, /opt, /opt/kaspersky, /usr/bin, /usr/lib, /usr/lib64.

Starting, restarting, and stopping Kaspersky Embedded Systems Security

To start the application, run the following command:

```
systemctl start kess
```

To stop the application, run the following command:

```
systemctl stop kess
```

To restart the application, run the following command:

```
systemctl restart kess
```

Monitoring the status of Kaspersky Embedded Systems Security

The Kaspersky Embedded Systems Security status is monitored by the watchdog service. The watchdog service is automatically launched when the application starts.

In the event of an application crash, a dump file is generated and the application is restarted automatically.

To export application settings, run the following command:

```
systemctl status kess
```

Viewing device protection status and application performance information

You can view information about the protection status of a device, as well as the status of Kaspersky Embedded Systems Security and its components on the device.

You can get information about the protection status of a device in the following ways:

- In the Web Console or in the Administration Console, using the statuses of the client devices (*OK*, *Critical*, *Warning*). The device on which Kaspersky Security Center Network Agent is installed is a client device for Kaspersky Security Center. The status of a client device can change to *Critical* or *Warning* for the following reasons:

  - In accordance with the rules defined in Kaspersky Security Center. For example, the status changes if a security application is not installed on the device, a virus scan has not been performed in a long time, application databases are outdated, the license has expired, or the application is unstable. For more details on the reasons for changing statuses and configuring conditions for assigning statuses, refer to the Kaspersky Security Center Help system.

  - Kaspersky Security Center receives the device status from the managed application, i.e., from Kaspersky Embedded Systems Security.

    > Receiving device status from a managed application must be enabled in Kaspersky Security Center in the lists of conditions for assigning the *Critical* and *Warning* statuses. Conditions for assigning device statuses are configured in the properties window of an administration group.

  For more details on client device statuses, refer to the Kaspersky Security Center Help system.

- In the Web Console or in the Administration Console, using the statuses of functional components of Kaspersky Embedded Systems Security on the device. In the properties of Kaspersky Embedded Systems Security installed on the device, a list of the functional components of the application is displayed. For each component, its status is displayed.

- On the command line using the following management commands:

  - The `kess-control --app-info` command, which displays information about application performance and the status of functional components and tasks of the application.

  - The `kess-control --supported-tech-info` command, which displays a list of application functions, information about their status (used or not used), and the Linux technologies that these application features are implemented with.

  - The `kess-control --get-statistic` command, which outputs application performance statistics and the list of mount points.

  - The `kess-control --export-metrics` command, which outputs operating system status metrics.

Viewing the protection status of a device in the Web Console

To view the protection status of a device in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to view information and click the device name.

4. In the properties window of the managed device that opens, on the **General** tab, select the **Protection** section.

The **Protection** section displays the following information about the device:

- **Visible in the network** is the visibility of the selected device in the network: *Yes* or *No*.
- **Device status** is the status of the client device generated based on the protection status criteria set by the administrator for the selected device and the device activity in the network: *OK*, *Critical*, or *Warning*.
- **Status description** represents the reasons for changing the status of the device to *Critical* or *Warning*.
- **Protection status** represents the current status of File Threat Protection on the selected device, such as *Running*, *Stopped*, or *Paused*.
- **Last full check** represents date and time when the last full scan task was completed on the selected device.
- **Viruses detected** represents a total number of malicious objects detected on the selected device (detected threat counter) since Kaspersky Embedded Systems Security was installed.
- **Objects that failed disinfection** represents a number of infected objects that Kaspersky Embedded Systems Security was unable to disinfect.

Viewing the protection status of a device in the Administration Console

To view the protection status of a device in the Administration Console:

1. In the Administration Console tree, in the **Managed devices** folder, select the administration group containing the necessary device.

2. In the workspace, select the **Devices** tab.

3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.

4. In the window that opens with the properties for the managed device, select the **Protection** section.

The **Protection** section displays the following information about the device:

- **Device status**: status of the client device generated based on the criteria set by the administrator for the protection status of the selected device and the device activity in the network.
- **All problems**: complete list of problems detected by the managed applications installed on the selected device. Each problem has a status that the application prompts to assign to the device.
- **Real-Time Protection status**: current status of File Threat Protection on the selected device, such as *Running* or *Stopped*. When the protection status changes, the new status is displayed in the device properties window only after the device is synchronized with the Administration Server.
- **Last on-demand scan**: date and time when the last malware scan was performed on the selected device.
- **Total threats detected**: total number of threats detected on the selected device since the installation of the application (first scan) or since the last reset of the threat counter.

  To reset the counter, click the **Reset** button.
- **Active threats**: the number of unprocessed files on the selected device.

Viewing information about the operation of an application in the Web Console

To view information about the application operation in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to view information and click the device name.

4. This opens a managed device properties window; in that window, go to the **Applications** tab.

5. In the list of applications installed on the device, click the name of the Kaspersky Embedded Systems Security application.

   The application properties window opens.

The **Kaspersky Embedded Systems Security 4.0 for Linux** window displays the following information about Kaspersky Embedded Systems Security:

- The **General** tab in the **Information** section displays general information about the installed application:
  - **Name** is the name of the application.
  - **Version** is the version number of the application.
  - **Installed** is the date and time when the application was installed on the device.
  - **Last software update**: date and time when Kaspersky Embedded Systems Security software modules were last updated.
  - **Last synchronization** is the date and time of the last connection of the device to the Kaspersky Security Center Administration Server.
  - **Current status**: status of File Threat Protection on the device, such as *Running* or *Paused*.
  - The **Installed updates** block contains information about installed versions of application updates.
  - Under **Application databases**, you can find information about the date and time of the application database update release and the date and time of the last update.
- On the **General** tab, the **Licenses** section contains information about license keys added to the application and the licenses corresponding to these keys.
- On the **General** tab, the **Components** section contains a list of functional components of the application. The status (for example, *Stopped, Suspended, Not Installed*) and version of each component is displayed.
- The **Events** tab displays a list of application events on the device.
- The **Event settings** tab displays the types of events that the application stores in event storage and how long they are stored.
- On the **Application settings** tab, you can do the following:
  - In the **Application Control** section, you can export configured component settings and application categories to a file, and import them from a file.
  - In the **Statistics** section, you can view application statistics and the list of mount points.

Viewing information about the operation of an application in the Administration Console

To view information about the application operation in the Kaspersky Security Center Administration Console:

1. In the Kaspersky Security Center Administration Console tree, in the **Managed devices** folder, select the administration group containing the required device.
2. In the workspace, select the **Devices** tab.

3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.

4. In the displayed window with the properties of the managed device, select the **Applications** section.

   The right part of the window displays a list of Kaspersky applications installed on the device.

5. Select Kaspersky Embedded Systems Security and double-click it to open the application properties window. Alternatively, you can use the **Properties** button in the lower part of the window.

   This opens the **Kaspersky Embedded Systems Security 4.0 for Linux settings** window.

The **Kaspersky Embedded Systems Security 4.0 for Linux settings** window displays the following information about Kaspersky Embedded Systems Security:

- The **General** section contains general information about the installed application:
  - **Version number**: the version number of the application.
  - **Installed** is the date and time when the application was installed on the device.
  - **Current status** is the status of File Threat Protection on the device, such as *Running* or *Paused*.
  - **Last software update**: date and time when Kaspersky Embedded Systems Security software modules were last updated.
  - **Installed updates** is the information about installed versions of application updates.
  - **Application databases** is the date and time when the application database update was released.
- In the **Application Control** section, you can export configured component settings and application categories to a file, and import them from a file.
- The **Statistics** section displays application statistics and information about mount points.
- The **Components** section contains a list of standard application components. The status (for example, *Stopped, Suspended, Not Installed*) and version of each component is displayed.
- The **License keys** section contains information about the active and reserve license keys.
- The **Event settings** section displays the types of events that the application stores in event storage and how long they are stored.
- The **Advanced** section contains information about the application management plug-in.

Viewing information about the operation of an application in the command line

To view information about the application, run the following command:

```
kess-control --app-info [--json]
```

where `--json`: output data in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

As a result of the command execution, the following information will be displayed in the console:

- **Name**. Application name.
- **Version**. Current application version.
- **Policy**. Information about whether a Kaspersky Security Center policy is applied on the device.
- Extended information about the policy applied on the device:
  - Policy name.
  - The date and time when the policy was last applied or updated on the device.
  - Kaspersky Security Center administration group to which the device belongs.
  - List of policy profiles applied on the device (if any).

  Extended policy information is displayed if the **Application is allowed to retrieve policy's extended data on device** setting of the Network Agent that is installed on the device allows retrieving of extended policy data from Kaspersky Embedded Systems Security. This setting is disabled by default. For details about Network Agent settings, please refer to the Kaspersky Security Center Help.
- Information about the main license under which the application is being used:
  - **Application license information** Application license information or application license key status.
  - **Kaspersky Embedded Systems Security license expiration date**. Application license expiration date and time (UTC).
- License information about the license that activates additional functionality of the application. These fields are displayed if the additional MDR functionality key has been added to the protected device:
  - **MDR license information**. Information about the license under which the Managed Detection and Response component is activated, or the status of the MDR license key.
  - **MDR license expiration date**. License expiration date and time (UTC).
- **Subscription status**. Subscription status. This field is displayed if the application is started under a subscription.
- **Backup state**. Information about the status of Backup.
- **Backup usage**. Backup size.
- **Last run date of the Scan_My_Computer task**. Time of the last Malware Scan task.
- **Last release date of databases**. Date and time the application databases were last released.
- **Application databases**. Information about whether the application databases were downloaded.

- **Using Kaspersky Security Network**. Information about using Kaspersky Security Network: `Extended KSN mode`, `Basic KSN mode` or `Disabled`.
- **Kaspersky Security Network infrastructure**. Information about the infrastructure solution used to work with Kaspersky reputation databases: `Kaspersky Security Network` or `Kaspersky Private Security Network`.
- **Integration with Kaspersky Managed Detection and Response**. Status of integration with Kaspersky Managed Detection and Response: `Enabled`, `Disabled`.
- **File Threat Protection**. Real-time File Threat Protection status.
- **System Integrity Monitoring**. System Integrity Monitoring component status.
- **Firewall Management**. Firewall Management component status.
- **Anti-Cryptor**. Anti-Cryptor component status.
- **Web Threat Protection**. Web Threat Protection component status.
- **Device Control**. Device Control component status.
- **Removable Drives Scan**. Removable Drives Scan component status.
- **BadUSB Attack Prevention**. BadUSB Attack Prevention component status.
- **Network Threat Protection**. Network Threat Protection component status.
- **Behavior Detection**. Behavior Detection component status.
- **Application Control**. Application Control component status.
- **Post-update actions**. Application update actions and the actions to be performed by the user.
- **Unstable application operation**. Information about application failure and dump file creation. This field is displayed if a failure occurred the last time the application was launched.

Viewing application statistics

To improve performance, you can exclude the files that are most frequently scanned by the File Threat Protection component, the paths to processes that are most frequently scanned by the Behavior Detection component, and the mount points detected on the device.

> On devices with operating systems that support fanotify, the most effective way to improve performance is to exclude mount points.

You can view statistics about the most frequently scanned files and started processes, as well as a list of detected mount points in the following ways:

- in the Web Console
- in the Administration Console
- in the command line

You can configure the parameters for how the application calculates and displays statistics in the kess.ini configuration file in the `[ScannerImpactStats]` section.

Viewing application statistics in the Web Console

To view application statistics in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to view information and click the device name.

4. This opens a managed device properties window; in that window, go to the **Applications** tab.

5. In the list of applications installed on the device, click the name of the Kaspersky Embedded Systems Security application.

6. In the application properties window that opens, on the **Application settings** tab, select **Statistics** → **Scan statistics**.

   The **Scan statistics** window will open.

   The **Scan statistics** window displays the following application statistics:

   - The **Most frequently scanned applications (File Threat Protection)** block displays the paths to the applications that the File Threat Protection component scans most frequently, and the number of times the component accesses these applications. Files are displayed in descending order of the number of times they are accessed.
   - Under **Most frequently started processes**, you can find paths to the processes that Kaspersky Embedded Systems Security detects most often and the number of such process starts. Paths to processes are displayed in descending order of the number of process starts.

   Using the **Export** buttons located at the top of each block, you can export the statistics displayed in that block to a text file.

Viewing application statistics in the Administration Console

To view application statistics in the Administration Console:

1. In the Kaspersky Security Center Administration Console tree, in the **Managed devices** folder, select the administration group containing the required device.

2. In the workspace, select the **Devices** tab.

3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.

4. In the window that opens with the properties of the managed device, select the **Applications** section.

   The right part of the window displays a list of Kaspersky applications installed on the device.

5. Select Kaspersky Embedded Systems Security and double-click it to open the application properties window. Alternatively, you can use the **Properties** button in the lower part of the window.

6. In the window that opens, select **Statistics** → **Scan statistics**.

   The following application statistics is displayed in the window on the right:

   - The **Most frequently scanned applications (File Threat Protection)** block displays the paths to the applications that the File Threat Protection component scans most frequently, and the number of times the component accesses these applications. Files are displayed in descending order of the number of times they are accessed.

   - Under **Most frequently started processes**, you can find paths to the processes that Kaspersky Embedded Systems Security detects most often and the number of such process starts. Paths to processes are displayed in descending order of the number of process starts.

   Using the **Export** buttons under each block, you can export the statistics displayed in that block to a text file.

Viewing a list of mount points in the Web Console

To view the list of mount points in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the necessary device. To do so, click the link in the **Current path** field above the list of managed devices and select an administration group in the window that opens.

   The list displays only the managed devices for the selected administration group.

3. In the list, find the device for which you want to view information and click the device name.

4. This opens a managed device properties window; in that window, go to the **Applications** tab.

5. In the list of applications installed on the device, click the name of the Kaspersky Embedded Systems Security application.

6. In the application properties window that opens, on the **Application settings** tab, select **Statistics** → **Mount points**.

   The **Mount points** window opens.

   The **Mount points** window displays a table with the following columns:

   - **Mount point path** is the path to the mount point detected on the device.
   - **Namespace** indicates whether the mount point is found in a namespace.
   - **Excluded** indicates whether the mount point has been added to scan exclusions.

   Mount point paths are sorted as follows:

   - Static, system, service and other mount points
   - Mounted snap packages and mount points in their namespaces

   By clicking the **Export** button, located at the top of the window above the list, you can export the list of mount points to a text file. The file contains a list of mount point paths, as well as the `[Namespace]` and `[Excluded]` tags.

Viewing the list of mount points in the Administration Console

To view the list of mount points in the Administration Console:

1. In the Kaspersky Security Center Administration Console tree, in the **Managed devices** folder, select the administration group containing the required device.
2. In the workspace, select the **Devices** tab.
3. In the list of managed devices, select the required device and double-click it to open the **Properties: <Task name>** window.
4. In the window that opens with the properties of the managed device, select the **Applications** section.

   The right part of the window displays a list of Kaspersky applications installed on the device.
5. Select Kaspersky Embedded Systems Security and double-click it to open the application properties window. Alternatively, you can use the **Properties** button in the lower part of the window.
6. In the window that opens, select **Statistics** → **Mount points**.

   The **List of mount points** table is displayed in the right part of the window and contains the following columns:

   - **Mount point path** is the path to the mount point detected on the device.
   - **Namespace** indicates whether the mount point is found in a namespace.
   - **Excluded** indicates whether the mount point has been added to scan exclusions.

Mount point paths are sorted as follows:

- Static, system, service and other mount points
- Mounted snap packages and mount points in their namespaces

By clicking the **Export** button below the list, you can export the list of mount points to a file in text format. The file contains a list of mount point paths, as well as the `[Namespace]` and `[Excluded]` tags.

Viewing application statistics and the list of mount points in the command line

To view application statistics and the list of mount points, run the following command:

```
kess-control [-S] --get-statistic [--files] [--processes] [--mountpoints]
```

where:

- `--files`: statistics of files most frequently scanned by the File Threat Protection component, and the number of times the component accesses these files.
- `--processes`: statistics of process starts most frequently detected by Kaspersky Embedded Systems Security, and the number of such starts.

  Excluding processes from scanning by the Behavior Detection component does not remove these processes from statistics, but it improves the performance of the Kaspersky Embedded Systems Security application.

- `--mountpoints`: list of mount points.

  The output includes the following information:

  - `[excluded]` indicates whether the mount point has been added to scan exclusions.
  - `[namespace]`, indicating whether the mount point is found in a namespace.
  - Path to the mount point found on the device.

  Mount point paths are sorted as follows:

  - Static, system, service and other mount points
  - Mounted snap packages and mount points in their namespaces

You can specify one or more options in any combination or no options at all. If you do not specify any options, the application displays three lists: statistics on the most frequently scanned files, statistics on the most frequently started processes, and the list of mount points found on the device.

Obtaining system performance metrics

Kaspersky Embedded Systems Security affects the operating system. To help you analyze the impact, the application can obtain metrics associated with application performance.

To configure the application to get metrics for operating system resources that are used by the application, run the following command:

```
kess-control [-J] --export-metrics [--period <interval between exports in seconds>|--interactive]
```

where:

- `--period` enables periodic output of results.
- `--interactive` enables interactive output (on the **Enter** key being pressed).

This command outputs the list of metrics to the console.

List of metrics

Internal name of the metric; label and type of the metric	Description and recommended threshold setting
lfs_system_descriptor_opened_count Sensor	Number of descriptors currently opened by the kess process. If the number of descriptors grows faster than 250 per hour, we recommend contacting Technical Support.
lfs_system_uptime_milliseconds Counter	Number of milliseconds from the startup of the operating system to now.
lfs_system_memory_usage_bytes 'type=virtual' label Sensor	Amount of virtual memory currently used by the kess process. We recommend taking steps to set a threshold based on the system information. You can get this information manually or using a script. For an example script, you can contact Technical Support.
lfs_system_memory_usage_bytes 'type=resident' label Sensor	Amount of resident memory currently used by the kess process. We recommend taking steps to set a threshold based on the system information. You can get this information manually or using a script. For an example script, you can contact Technical Support.
lfs_system_memory_usage_bytes 'type=swap' label Sensor	Amount of swap memory currently used by the kess process. We recommend taking steps to set a threshold based on the system information. You can get this information manually or using a script. For an example script, you can contact Technical Support.
lfs_system_cpu_usage_milliseconds 'type=user' label Counter	Milliseconds of CPU time used by the kess process in user space, from the start of the kess process to now. We recommend taking steps to set a threshold based on the system information. You can get this information manually or using a script. For an example script, you can contact Technical Support.
lfs_system_cpu_usage_milliseconds 'type=kernel' label Counter	Milliseconds of CPU time used by the kess process in kernel space, from the start of the kess process to now. We recommend taking steps to set a threshold based on the system information. You can get this information manually or using a script. For an example script, you can contact Technical Support.
lfs_system_cpu_usage_milliseconds 'type=total' label Counter	Milliseconds of CPU time used by the kess process in user space and kernel space, from the start of the kess process to now. If the CPU time used by the kess process exceeds the rest of the workload, we recommend contacting Technical Support.

Internal name of the metric; label and type of the metric	Description and recommended threshold setting
lfs_tcpSynInterceptor_connection_hanging_count Counter	Total number of network connections hanging at the handshake stage in the TCP network between the server and the client from the start of the kess process to now (network connections are considered hanging if they have been waiting for the remote server to allow the connection for a time between 1 second to 2 minutes, depending on the system setting in /proc/sys/net/ipv4/tcp_syn_retries). An increase of this metric may be caused by the remote server being inaccessible or missing. If the number of hanging network connections grows by more than 10 per minute, we recommend contacting Technical Support.
lfs_tcpSynInterceptor_connection_count Sensor	Number of network connections currently waiting to connect to the remote server.
lfs_tcpSynInterceptor_verdict_count "verdict=allow" label Counter	Total number of intercepted connections that have been established from the start of the kess process to the present.
lfs_tcpSynInterceptor_verdict_count 'verdict=drop' label Counter	Total number of intercepted connections that have not been established, counting from the start of the kess process to now.
lfs_tcpSynInterceptor_verdict_count 'verdict=unknown' label Counter	Total number of network connections from the start of the kess process to the present moment with a solution of the remote server that cannot be processed.
lfs_tcpSynInterceptor_verdict_latency_milliseconds Histogram	Histogram of the time the remote server takes to make a solution on a connection from the time when the SYN packet is intercepted to the time when resolution is issued for it.
lfs_tproxy_connection_count 'direction=inbound' label Sensor	Current number of intercepted inbound network connections.
lfs_tproxy_connection_count 'direction=outbound' label Sensor	Current number of intercepted outbound network connections.
lfs_tproxy_connection_orphaned_count Counter	The total number of connections (from the start of Web Threat Protection and Network Threat Protection protection components) intercepted by Kaspersky Embedded Systems Security despite their interception not being intended (for example, due to an incompatible iptables configuration or the fact that the application did not have time to process a new interception configuration). Such connections are rejected. If the total number of such connections exceeds 10, which may indicate incorrect operation of the application or the system, we recommend contacting Technical Support.
lfs_tproxy_threadPool_task_count Counter	Total number of tasks (new connection, new batch of data, data sent, and so on) that have been added to the thread pool since the start of the Web Threat Protection and Network Threat Protection components. If the total number of created but uncompleted tasks keeps growing, you need to add resources, configure exclusions, or contact Technical Support.
lfs_tproxy_threadPool_task_duration_milliseconds Histogram	Histogram of the execution time of tasks that process intercepted connections. If the execution time of 95% of tasks exceeds 1 second, you need to add resources, configure exclusions, or contact Technical Support.
lfs_tproxy_socket_count Sensor	Current number of sockets (listening sockets and connection sockets). An intercepted connection has 2 connection sockets.
lfs_trafficScanning_object_hanging_count Sensor	Current number of hanging object scans in intercepted traffic (scans with a duration over 1 minute are considered hanging). The total number of hanging object scans exceeding 35 indicates a problem with the application databases. In such cases, we recommend contacting Technical Support.

Internal name of the metric; label and type of the metric	Description and recommended threshold setting
lfs_trafficScanning_object_duration_milliseconds Histogram	Histogram of durations of intercepted traffic object scans. If the scan time for 95% of objects exceeds 1 second, you need to add resources, update application databases, or contact Technical Support. If the growth rate of packet scans in the intercepted traffic is 0 and the value of the lfs_tproxy_threadPool_task_count metric is not 0, you need to update the application databases or contact Technical Support.
lfs_fileMonitor_cache_size Sensor	Current size of the File Monitor (file revision cache), which monitors file modification. If the current File Monitor cache size is greater than 50,000, we recommend contacting Technical Support.
lfs_fileMonitor_cache_hit_count Counter	Hit count of the File Monitor cache (file revision cache), which monitors file modification.
lfs_fileMonitor_cache_miss_count Counter	Miss count of the File Monitor cache (file revision cache), which monitors file modification.
lfs_faCache_file_count Sensor	Current size of the cache of fanotify interceptor files that do not need to be scanned. If the current file cache size is greater than 100,000, we recommend contacting Technical Support.
lfs_faCache_file_hit_count Counter	Hit count of the cache of fanotify interceptor files that do not need to be scanned.
lfs_faCache_file_miss_count Counter	Miss count of the cache of fanotify interceptor files that do not need to be scanned.
lfs_faCache_volume_count Sensor	Current size of the fanotify interceptor mount point cache.
lfs_oas_cache_file_count Sensor	Current size of the cache of File Threat Protection files that do not need to be scanned. If the current file cache size is greater than 100,000, we recommend contacting Technical Support.
lfs_oas_cache_file_hit_count Counter	Hit count of the cache of File Threat Protection files that do not need to be scanned.
lfs_oas_cache_file_miss_count Counter	Miss count of the cache of File Threat Protection files that do not need to be scanned.
lfs_oas_cache_volume_count Sensor	Current size of the File Threat Protection mount point cache.
lfs_processManager_cache_size Sensor	Current number of active processes in the system that are cached.
lfs_processManager_cache_hit_count Counter	Hit count for the system process cache.
lfs_processManager_cache_miss_count Counter	Miss count for the system process cache.
lfs_blockingProcessInterceptor_cache_size Sensor	Current size of the cache of interpreters that can be used to start processes. This number depends on how intensely scripts are run on a specific device. If the current file cache size is greater than 10000, we recommend contacting Technical Support.
lfs_blockingProcessInterceptor_cache_hit_count Counter	Hit count of the interpreter cache.
lfs_blockingProcessInterceptor_cache_miss_count Counter	Miss count of the interpreter cache.
lfs_prevention_cache_file_count Sensor	Current size of the "Execution prevention for objects" task's file cache for objects that do not need scanning.
lfs_prevention_cache_file_hit_count Counter	Number of hits in the "Execution prevention for objects" task's file cache for objects that do not need scanning.

Internal name of the metric; label and type of the metric	Description and recommended threshold setting
lfs_prevention_cache_file_miss_count Counter	Number of misses in the "Execution prevention for objects" task's file cache for objects that do not need scanning.
lfs_prevention_cache_volume_count Sensor	Current size of the "Execution prevention for objects" task's mount point cache.
eka_telemetry_metrics_registry_callbacks_exec_duration_milliseconds Sensor	How long the current metrics update took for all clients registered in the registry (in milliseconds).
eka_telemetry_metrics_registry_registered_callbacks Sensor	Current number of metric update callbacks that the MetricsRegistry has performed for all registered clients.
eka_telemetry_metrics_registry_scrape_duration_milliseconds Sensor	How long it took to get all registry metrics (in milliseconds). If 99% of the values of this metric exceed 1 second for a 10-minute period, we recommend contacting Technical Support.

We recommend analyzing metrics in data visualization systems, for example, Grafana.

You can use a script to get information about your operating system and device. For an example script and integration with the Grafana system, you can contact Technical Support.

You can publish exported metrics to monitoring systems such as Prometheus and Zabbix. To integrate with monitoring systems, you can use a script that gets information from the application and publishes it to the monitoring system. To get the script, you can contact Technical Support.

Updating application databases and modules

The update functionality (including anti-virus signature updates and code base updates) may not be available in the application in the territory of the USA.

Updating the databases and application modules of Kaspersky Embedded Systems Security ensures up-to-date protection on your device. New viruses, malware, and other types of threats appear worldwide on a daily basis. The application databases contain information about the threats and the ways to neutralize them. To detect threats quickly, you are urged to regularly update the application databases and modules.

Current application license is required for regular database updates. If there is no current license, you will only be able to perform one update.

During the update process, the databases and application modules are downloaded and installed on your device. You can obtain updates for databases and application modules from Kaspersky update servers, from the Administration Server repository, from local or network directories, and from other update sources.

During an update, the application modules and databases on your device are compared with the up-to-date version at the update source. If your current databases and application modules differ from their respective up-to-date versions, the missing portions of the updates will be installed on your device.

If the databases are obsolete, the update package may be large, which may cause additional Internet traffic (up to several dozen MB). The amount of the disk space can be up to 3 GB.

Updates are downloaded from Kaspersky update servers or from other FTP, HTTP, or HTTPS servers over standard network protocols. By default, Internet connection settings are determined automatically. If you are using a proxy server, specify the proxy server settings in the general settings of the application.

Regardless of the update source, the update package is downloaded and the database and application module updates are installed on the device using the *Update* task. An *Update* predefined task is created in the application. Using this task, you can perform scheduled and on-demand updates of databases and application modules and configure update settings.

You can use the *Update* group task, which the Kaspersky Security Center Initial Configuration Wizard automatically creates after installing the Kaspersky Embedded Systems Security administration MMC plug-in or administration web plug-in. You can also create update user tasks in the command line and in Kaspersky Security Center.

You can configure the following settings for updating databases and application modules:

- Select the source from which the application will receive updates, depending on the update scenario used.
- Configure the response timeout of a selected update source when attempting to connect to it. If an update source does not respond within the specified time, the application contacts the next update source in the list.
- Configure the task run schedule for updates. By default, the application updates the databases once every 60 minutes.

Updating databases and modules

During an update, the following objects are downloaded and installed on your device:

- Application databases. Application databases include databases of malware signatures, a description of network attacks, databases of malicious and phishing web addresses, databases of banners, spam databases, and other data.

  If the database update on the device is interrupted or finishes with an error, the application continues to use the previously installed database version. If application databases were not installed before, the application continues functioning in "without databases" mode. Database and application module updates are still available.

  If the database update is successful, but the databases themselves are corrupted and the application terminates with an error several times in a row, the databases are automatically deleted. The application continues to work in the "without databases" mode; the database and application module update functionality remains available.

  The database is up-to-date if it was downloaded less than three days ago. By default, the application generates the *Application databases are out of date* event if the last installed database updates were published on the Kaspersky servers more than three but less than seven days ago. If the databases have not been updated for seven days, the application generates the *Application databases are totally out of date* event.

- Application modules. Module updates are intended to eliminate vulnerabilities in the application and to improve methods of protecting devices. Module updates may change the behavior of application components and add new capabilities.

  The application module can be installed regardless of the state of the application (started or stopped, managed by a Kaspersky Security Center policy) and the update schedule. Kaspersky Embedded Systems Security continues protecting your device during the application module update procedure. During the update, application settings and the application log file are migrated to the new version of the application.

  If the transfer of application settings fails for any reason, the application is set to the default values.

  Changes to the application settings made after the update is complete and before the application restarts are not saved.

  If the application does not work properly after the update, it automatically rolls back to the previous version. It is recommended to contact Kaspersky Technical Support.

Updating sources and update scenarios

An *update source* is a resource that contains updates for Kaspersky Embedded Systems Security databases and application modules. Update sources can be FTP, HTTP, or HTTPS servers (such as Kaspersky update servers), as well as local or network directories mounted by the user.

The main application update sources are Kaspersky update servers. You can specify other update sources in the *Update* task settings. If an update cannot be performed from an update source, Kaspersky Embedded Systems Security switches to the next update source.

Kaspersky Embedded Systems Security supports the following scenarios for updating databases and application modules:

- Update from Kaspersky update servers. Kaspersky update servers are located in different countries around the world, which ensures a high reliability of updates. If an update cannot be performed from one server, the application switches over to the next server. Updates are downloaded via HTTPS protocol.

- Centralized update Centralized update reduces external Internet traffic, and provides for convenient monitoring of the update.

  Centralized update consists of the following steps:

  1. Download the update package to a repository within the organization's network.

     You can use the repository of the Kaspersky Security Center Administration Server as the repository.

     The update package is downloaded to the Administration Server repository via the *Download updates to Administration Server repository* task of the Administration Server.

     If you manage the application using Kaspersky Security Center Cloud Console, you can use the repositories of the distribution points (devices with Network Agent installed) as the repository. For more details about distribution points, refer to Kaspersky Security Center Help.

  2. Distribute the update package to client devices

     The update package is distributed to the client devices by the *Update* task of Kaspersky Embedded Systems Security. In the task settings, select Kaspersky Security Center as the update source.

- Updating from a local or network directory (SMB/NFS) mounted by a user, or from an FTP, HTTP, or HTTPS server. You can specify a custom update source in *Update* task settings. When choosing your own HTTPS server as the update source, you need to make sure that it has the correct certificate chain. A self-signed certificate can only be used as the root certificate in a certificate chain. In this case, you need to add it to the list of trusted certificates on the device.

Updating application databases and modules in the Web Console

In the Web Console, you can update databases and application modules using the *Update* task. You can use the automatically created *Update* group task, as well as create user tasks for updating.

To configure update settings in the Web Console:

1. In the main window of the Web Console, select **Assets (Devices)** → **Tasks**.

   The list of tasks opens.

2. Do one of the following:

   - If you want to edit the settings of a task that is run on all devices included in a specific administration group, click the link in the **Current path** field in the upper part of the window and select the administration group in the window that opens.

     The list displays only tasks configured for the selected administration group.

- If you want to edit the settings of a task that is run on one or multiple devices (a task for a set of devices), click the link in the **Current path** field in the upper part of the window and select the top node with the name of the Administration Server in the window that opens.

  The list displays all tasks created on the Administration Server.

3. In the list of tasks, select the required **Update** task and open the task properties window by clicking the link in the task name.

4. In the task properties window, select **Application settings** tab. Select the **Update sources** section in the list on the left.

5. In the table, select the update source from which the application will receive updates for databases and modules, depending on the update scenario used.

   The list of update sources contains two sources: Kaspersky update servers and Kaspersky Security Center. The **Status** column indicates whether the application is using this update source when running the task. You can add other update sources to the list by clicking the **Add** button. You can specify FTP-, HTTP-, or HTTPS servers as update sources. If an update cannot be performed from an update source, Kaspersky Embedded Systems Security switches to the next update source. The application accesses update sources in the order in which they appear in the table.

6. Select the **Settings** section and configure other update settings.

7. Select the **Schedule** tab and configure the schedule for running the update task.

   If you have selected **Kaspersky Security Center** as the update source, select **When downloading updates to the repository** from the **Start task** drop-down list. For more details about scheduling tasks, refer to the Kaspersky Security Center Help system.

8. Click the **Save** button to save the changes made.

The task will start according to the configured schedule. You can also run the task manually.

Update sources for the Update task section

Setting	Description

Table of update sources	This table contains a list of database update sources. During the update process, the application accesses update sources in the order they appear in the table.
	The table contains the following columns:
	• **Update source** is the address of the update source:
	  • **Kaspersky update servers**, where database updates for Kaspersky applications are published (default value).
	  • **Kaspersky Security Center** – Kaspersky Security Center Administration Server.
	  • Other sources from the local or global network added by the user: HTTP, HTTPS, or FTP servers or directories on local network servers.
	• **Status** indicates whether the source is used in the task (**Enabled** or **Disabled**). You can change the status by enabling or disabling the **Use this source** toggle switch in the **Update source** window, which you can open by clicking the link with the name of the source.
	You can **add**, **edit**, **delete**, **move up**, or **move down** update sources in the table.
	Clicking the **Move down** button moves the selected item down in the table.
	This button is available if only one item is selected in the table.
	Clicking the **Move up** button moves the selected item up in the table.
	This button is available if only one item is selected in the table.
	Clicking the **Delete** button removes the selected item from the table.
	This button is available if at least one item is selected in the table.
	The selected element's settings are changed in a separate window.
	Clicking the **Add** button opens a window where you can specify the new item settings.

Update task settings section

Setting	Description
Timeout for response from the update source (sec)	The maximum period of time that the application waits for a response from the selected update source (in seconds). When no response has arrived by this time, an event involving a loss of communication with the update source is logged in the task log. Possible values: 0–120. If 0 is specified, the period of time that the application waits for a response from the selected source is unlimited. Default value: 10 seconds.
Application update download mode	In the drop-down list, you can select how you want the automatically installed updates (autopatches) to be downloaded: • **Do not download**. When this item is selected, autopatches are not automatically downloaded. • **Download only**. When this list item is selected, autopatches are automatically downloaded, but not installed on client devices (default setting). • **Download and install**. When this list item is selected, autopatches are automatically downloaded and installed on client devices. After an autopatch is installed, the application restarts automatically.

Updating application databases and modules in the Administration Console

In the Administration Console, you can update databases and application modules using the *Update* task. You can use the automatically created *Update* group task, as well as create user tasks for updating.

To configure update settings in the Administration Console:

1. In the Administration Console, perform one of the following actions:

   - To edit the settings of a task that is run on devices included in the specified administration group, select this administration group in the console tree, then select the **Tasks** tab in the workspace.

   - To edit the settings of a task that is run on one or multiple devices (a task for a set of devices), select the **Tasks** folder in the console tree.

2. In the list of tasks, select the required **Update** task and double-click it to open the task properties window.

3. In the task properties window, select the **Update sources** section in the list on the left.

4. In the table, select the update source from which the application will receive updates for databases and modules, depending on the update scenario used.

   The list of update sources contains two sources: Kaspersky update servers and Kaspersky Security Center. The **Status** column indicates whether the application is using this update source when running the task. You can add other update sources to the list by clicking the **Add** button. You can specify FTP-, HTTP-, or HTTPS servers as update sources. If an update cannot be performed from an update source, Kaspersky Embedded Systems Security switches to the next update source. The application accesses update sources in the order in which they appear in the table.

5. Select the **Settings** section and configure other update settings.

6. Select the **Schedule** section and configure the schedule for running the update task.

   If you have selected **Kaspersky Security Center** as the update source, select **When downloading updates to the repository** from the **Scheduled start** drop-down list. For more details about scheduling tasks, refer to the Kaspersky Security Center Help system.

7. Click **Apply** or **OK** in the **Properties: <Task name>** window to save the changes made.

The task will start according to the configured schedule. You can also run the task manually.

Update sources for the Update task section

Setting	Description

Update source	This table contains a list of database update sources. During the update process, the application accesses update sources in the order they appear in the table.
	The table contains the following columns:
	• **Source address** is the address of the update source:
	  • **Kaspersky update servers**, where database updates for Kaspersky applications are published (default value).
	  • **Kaspersky Security Center** – Kaspersky Security Center Administration Server.
	  • Other sources from the local or global network added by the user: HTTP, HTTPS, or FTP servers or directories on local network servers.
	• **Status** indicates whether the update source is used in the task (**In use** or **Not in use**). You can change the status by selecting or clearing the **Use this source** check box in the **Update source** window that opens when you click the **Edit** button.
	You can **add**, **edit**, **delete**, **move up**, or **move down** update sources in the table.
	Clicking the **Move down** button moves the selected item down in the table.
	This button is available if only one item is selected in the table.
	Clicking the **Move up** button moves the selected item up in the table.
	This button is available if only one item is selected in the table.
	Clicking the **Delete** button removes the selected item from the table.
	This button is available if at least one item is selected in the table.
	The selected element's settings are changed in a separate window.
	Clicking the **Add** button opens a window where you can specify the new item settings.

Update task settings section

Setting	Description
Timeout for response from the update source (sec)	The maximum period of time that the application waits for a response from the selected update source (in seconds). When no response has arrived by this time, an event involving a loss of communication with the update source is logged in the task log. Possible values: 0–120. If 0 is specified, the period of time that the application waits for a response from the selected source is unlimited. Default value: 10 seconds.
Application update download mode	In the drop-down list, you can select how you want the automatically installed updates (autopatches) to be downloaded: • **Do not download**. When this item is selected, autopatches are not automatically downloaded. • **Download only**. When this list item is selected, autopatches are automatically downloaded, but not installed on client devices (default setting). • **Download and install**. When this list item is selected, autopatches are automatically downloaded and installed on client devices. After an autopatch is installed, the application restarts automatically.

Updating application databases and modules in the command line

On the command line, you can update databases and application modules in the following ways:

- Using the *Update* predefined task. You can manually start, stop, pause, or resume this task and configure the task run schedule. You can manage update settings by editing the settings of this task.
- Using user tasks for updating (tasks of the *Update* type). You can manually start user tasks and configure the task schedule.

Update task settings

Setting	Description	Values
`ConnectionTimeout`	Response timeout (in seconds) of an update source while attempting to connect to it. If an update source does not respond within the specified time interval, the application contacts the next update source in the list.	You can use only integers within the range from 0 to 120. Default value: 10.
`ApplicationUpdateMode`	How automatically installed application updates (autopatches) are downloaded and installed.	`Disabled`: do not download or install autopatches. `DownloadOnly` (default): download autopatches, but do not install them. `DownloadAndInstall`: automatically download and install autopatches. After an autopatch is installed, the application restarts automatically.
The **[Sources.item_#]** section contains the following settings:		
`Enabled`	Use of the update source specified in the `SourceType` setting. To run the task, at least one update source needs to be enabled.	`Yes` (default) – the application is using the update source. `No` – the application is not using the update source.
`SourceType`	Source from which the application receives updates.	`KLServers` (default value) — The application receives updates from one of the Kaspersky update servers. Updates are downloaded via HTTPS protocol. `SCServer`. The application downloads updates to the protected device from the Administration Server installed on the local network. You can select this update source if you use Kaspersky Security Center for centralized administration of device protection in your organization. `Custom`. The application downloads updates from a custom source specified in the `URL` setting. You can specify directories on FTP, HTTP, and HTTPS servers or directories on any device mounted on the protected client device, including directories on remote devices mounted via the Samba or NFS protocols.
`URL`	Address of the custom update source on the local area network or the Internet.	The default value is not defined. Examples: `URL=http://example.com/bases/` – address of the HTTP server with the directory that contains updates. `URL=/home/bases/` – directory on the protected computer that contains application databases.

Updating using Kaspersky Update Utility

To reduce Internet traffic, you can configure updates of application databases and modules on devices of the organization's LAN from a shared directory by using the Kaspersky Update Utility. For this purpose, one of the devices in the organization's LAN must receive update packages from the Kaspersky Security Center Administration Server or from Kaspersky update servers and use the utility to copy the received update packages to the shared directory. Other devices on the organization's LAN will be able to receive the update package from this shared directory.

To configure Kaspersky Security Center to update databases from a shared directory using the Kaspersky Update Utility:

1. Install Kaspersky Update Utility on one of the devices of the organization's LAN.

   You can download the Kaspersky Update Utility distribution kit from the Kaspersky Technical Support website.

2. Configure copying of the update package to the shared directory in the Kaspersky Update Utility settings.

   Select the update source (for example, the Administration Server repository) and the shared directory to which the Kaspersky Update Utility will copy update packages. For detailed information about using Kaspersky Update Utility, refer to Kaspersky Knowledge Base.

3. Configure updates of application databases and modules from the specified shared directory on other devices in the organization's local network:

   a. Open the properties of the **Update** task that will be performed on the required device using the Web Console or using the Administration Console.

   b. In the task properties, go to the **Update sources** section.

   c. In the **Update sources** section, select the **Other sources on the local or global network** option.

   d. In the table of update sources, click the **Add** button and specify the path to the shared directory.

   > The address of the update source must match the address specified in the Kaspersky Update Utility settings.

   e. Select the **Use this source** check box and click **OK**.

   f. In the table, set the order of the update sources using the **Up** and **Down** buttons.

   g. Save the changes to the task settings.

To configure Kaspersky Security Center to update databases from a shared directory using the command line:

1. Install Kaspersky Update Utility on one of the devices of the organization's LAN.

   You can download the Kaspersky Update Utility distribution kit from the Kaspersky Technical Support website.

2. Configure copying of the update package to the shared directory in the Kaspersky Update Utility settings.

   Select the update source (for example, the Administration Server repository) and the shared directory to which the Kaspersky Update Utility will copy update packages. For detailed information about using Kaspersky Update Utility, refer to Kaspersky Knowledge Base.

3. Configure updates of application databases and modules from the specified shared directory on other devices in the organization's local network: in the `Update` task settings, set `SourceType=Custom` and specify the path to the shared directory in the `[CustomSources.item_#]` section.

   > The address of the update source must match the address specified in the Kaspersky Update Utility settings.

Rolling back application database and module updates

After the application databases are updated for the first time, the rollback of the application databases to their previous versions becomes available.

Every time a user starts the update process, Kaspersky Embedded Systems Security creates a backup copy of the current application databases. This allows you to roll back the application databases to a previous version if needed.

Rolling back the last database update may be useful, for example, if the new application database version contains invalid signatures, which causes Kaspersky Embedded Systems Security to block safe applications.

In the command line, to roll back updates, you can run the *Rollback* predefined task or create and run user tasks for rolling back updates (tasks of the *Rollback* type).

In Kaspersky Security Center, you can create rollback tasks for administration groups or for individual devices using the Web Console or the Administration Console.

The *rollback* task does not have any settings.

File Threat Protection

File Threat Protection component prevents infection of the device file system. The component is enabled automatically with the default settings when Kaspersky Embedded Systems Security starts. It resides in the device operating memory and scans all files that are opened, saved, and launched in real time.

Upon detecting malware, Kaspersky Embedded Systems Security may remove the infected file and terminate the malware process started from this file.

The operation of the component is affected by the file operation interception mode, which you can select in the general settings of the application. By default, access to the file is blocked for the duration of the scan.

You can enable or disable File Threat Protection, and also configure the protection settings:

- Select the file scan mode (when opened, or when opened and modified).
- Enable or disable scanning of archives, mail databases, email messages in text format.
- Temporarily exclude files in text format from rescans.
- Limit the size of an object to be scanned and the duration of the object scan.
- Select the actions to be performed by the application on the infected objects.
- Configure the *scan scopes*. The application will scan objects in the specified area of the file system.
- Configure exclusions of objects from scans. *Scan exclusion* is a set of conditions. When these conditions are met, the application does not scan the objects for viruses and other malware. You can exclude from scans:
  - Objects by name or mask
  - Objects by the name of the threats detected in them
  - Files and directories in specified areas of the file system
  - Processes and files being modified by the specified process

  The application also automatically excludes some internal processes from the scan, for example, the process of the Watchdog server.
- Configure the use of the heuristic analyzer and iChecker technology during a scan.
- Enable or disable the logging of information about scanned non-infected objects, about scanning objects in archives, and about unprocessed objects.

To optimize the File Threat Protection component, you can exclude from scans any files being copied from network directories. Files are scanned only after the process of copying to a local directory is finished. To exclude files located in network directories from scans, configure exclusion based on processes for the utility used for copying from network directories (for example, for the `cp` utility). If you manage the application using Kaspersky Security Center, you can configure exclusion based on processes in the Web Console or the Administration Console. If you are administering the application using the command line, you can configure an exclusion by process by adding an `[ExcludedForProgram.item_#]` section to the settings of the OAS task.

Configuring File Threat Protection in the Web Console

In the Web Console, you can manage File Threat Protection settings in the policy properties (**Application settings → Host Threat Protection → File Threat Protection**).

File Threat Protection component settings

Setting	Description
Scan exclusions	A *scan exclusion* is a set of conditions that must be fulfilled so that Kaspersky Embedded Systems Security will not scan a particular object for viruses and other threatening applications. Scan exclusions let you work with legitimate applications that intruders can use to compromise the computer or user data. Such applications do not necessarily have a malicious purpose, but such applications can be exploited by hackers. For detailed information about legitimate applications that intruders can use to compromise the computer or user data, please visit the Kaspersky Encyclopedia website. Kaspersky Embedded Systems Security supports environment variables and `*` and `?` characters in masks.
Action on threat detection	**Disinfect.Delete if disinfection is not possible**. If this action is selected, the application automatically attempts to disinfect the detected infected file. If disinfection proves impossible, the application deletes the file. **Disinfect.Block if disinfection fails**. If this action is selected, the application automatically attempts to disinfect all detected infected files. If disinfection fails, the application blocks these files (read/write). **Block**. If this action is selected, the application automatically blocks infected files without attempting to disinfect them. Before disinfecting or deleting an infected file, the application creates a backup copy of the file in case it needs to be restored later or disinfection becomes possible.
Protection scope	Contains objects that are scanned by the File Threat Protection component. A scan object may be a hard drive, removable drive, network drive, folder, file, or multiple files defined by a mask. By default, the File Threat Protection component scans files that are started on any hard drives, removable drives, or network drives. The protection scope for these objects cannot be changed or deleted. You can only exclude an object (such as removable drives) from scans. For Linux devices, you can modify or delete scan scopes of the File Threat Protection component. Specify at least one protection scope for the component to work.
Scanning compound files	A common technique of concealing viruses and other malware is to implant them in compound files, such as archives or databases. To detect viruses and other threats concealed in this way, the compound file must be unpacked, which can slow down the scan. You can limit the types of compound files to be scanned and thereby speed up scanning. • **Archives**. This includes the scanning of ZIP, GZIP, BZIP, RAR, TAR, ARJ, CAB, LHA, JAR, ICE and other types of archives. The application identifies archives not just by extension, but also by format. When scanning archives, the application performs recursive unpacking. This makes it possible to detect threats in nested archives (archive inside an archive). • **Distribution packages and self-extracting archives**. Scanning installation packages of third-party applications and self-extracting archives. Self-extracting archives are archives that contain an executable extraction module. To scan self-extracting archives, the **Scan archives** check box must be selected. • **Mail format files**. Scan plain text email messages.
Do not scan archives larger than (MB)	If this check box is selected, the application does not scan compound files if their size exceeds the specified value. If this check box is cleared, the application scans compound files of all sizes. The application scans large files that are extracted from archives regardless of whether the check box is selected or not.
Limit object scan time to (sec)	If the check box is selected, the application stops scanning compound files after the specified time elapses. If the value is set to `0`, the scan time is unlimited.
Block access to files during scans	If the check box is selected, the application blocks access to files on the device for the duration of the scan by the File Threat Protection and Device Control components.

Configuring File Threat Protection in the Administration Console

In the Administration Console, you can manage File Threat Protection settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → File Threat Protection**).

File Threat Protection component settings

Setting	Description
Enable File Threat Protection	This check box enables or disables File Threat Protection component on all managed devices. This check box is selected by default.
File Threat Protection mode	In this drop-down list, you can select the File Threat Protection component mode: • **Smart check** (default value) – scan a file when there is an attempt to open it and scan it again when there is an attempt to close it if the file has been modified. If a process accesses and modifies a file multiple times in a certain period, the application scans the file again only when the process closes it for the last time. • **When opened** – scan the file on an attempt to open it for reading, execution, or modification. • **When opened and modified** – scan a file on an attempt to open it, and scan it again on an attempt to close it if the file has been modified.
Scan	This group of settings contains buttons that open windows where you can configure the scan scopes and scan settings.
Actions on threat detection	This group of settings contains the **Configure** button. Clicking this button opens the **Actions on threat detection** window, where you can configure the actions that the application performs on detected infected objects.

Scan scopes window

The table contains the scan scopes. The application will scan files and directories located in the paths specified in the table. By default, the table contains one scan scope that includes all directories of the local file system.

Scan scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Scope name	Scan scope name.
Path	Path to the directory that the application scans.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure scan scopes.

Scan scope settings

Setting	Description
Scan scope name	Field for entering the scan scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this scan scope. If this check box is cleared, the application does not process this scan scope. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	The settings block lets you set the scan scope. You can select the file system type in the drop-down list of file systems: • **Local** (default value) – local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
	If **Shared** or **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want to add to the scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default – the application scans all directories of the local file system. If the **Local** type is selected in the drop-down list of file systems, and the path is not specified, the application scans all directories of the local file system.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the scan scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Scan settings window

In this window, you can configure file scan settings while File Threat Protection is enabled.

File Threat Protection settings

Setting	Description
Scan archives	This check box enables or disables scan of archives. If this check box is selected, Kaspersky Embedded Systems Security scans archives. To scan an archive, the application has to unpack it first, which may slow down scanning. You can reduce the duration of archive scans by enabling and configuring the **Skip file that is scanned for longer than (sec)** and **Skip file larger than (MB)** settings in the **General scan settings** section. If this check box is cleared, Kaspersky Embedded Systems Security does not scan archives. This check box is cleared by default.
Scan SFX archives	This check box enables or disables *self-extracting archive* scans. Self-extracting archives are archives that contain an executable extraction module. If this check box is selected, Kaspersky Embedded Systems Security scans self-extracting archives. If this check box is cleared, Kaspersky Embedded Systems Security does not scan self-extracting archives. This check box is available if the **Scan archives** check box is unchecked. This check box is cleared by default.
Scan mail databases	This check box enables or disables scans of mail databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail applications. If this check box is selected, Kaspersky Embedded Systems Security scans mail database files. If this check box is cleared, Kaspersky Embedded Systems Security does not scan mail database files. This check box is cleared by default.
Scan mail format files	This check box enables or disables scan of files of plain-text email messages. If this check box is selected, Kaspersky Embedded Systems Security scans plain-text messages. If this check box is cleared, Kaspersky Embedded Systems Security does not scan plain-text messages. This check box is cleared by default.
Skip text files	Temporary exclusion of files in text format from scans. If the checkbox is selected, Kaspersky Embedded Systems Security does not scan text files if they are reused by the same process for 10 minutes after the most recent scan. This setting makes it possible to optimize scans of application logs. If this check box is unselected, Kaspersky Embedded Systems Security scans text files. This check box is cleared by default. This setting is applied if you use system event interception mechanism based on fanotify. When using an updatable kernel module, the scan is optimized by using the scan result cache.
Skip database journal files	Temporarily excluding database journal files from scanning. If this check box is selected, the application does not scan database journal files if the journal file is reused by the same process within 10 minutes after the most recent scan. This setting makes it possible to optimize scans of database journal files. If this check box is cleared, the application scans database journal files. This check box is cleared by default. This setting is applied if you use system event interception mechanism based on fanotify. When using an updatable kernel module, the scan is optimized by using the scan result cache.
Skip file that is scanned for longer than (sec)	In this field, you can specify the maximum time to scan a file, in seconds. After the specified time elapses, Kaspersky Embedded Systems Security stops scanning the object. Possible values: 0–9999. If the value is set to 0, the scan time is unlimited. Default value: 60.
Skip file larger than (MB)	In this field, you can specify the maximum size of a file to scan, in megabytes. Possible values: 0–999999. If the value is set to 0, Kaspersky Embedded Systems Security scans files of any size. Default value: 0.

Setting	Description
Log clean objects	This check box enables or disables the logging of *ObjectProcessed* type events. If this check box is selected, Kaspersky Embedded Systems Security logs *ObjectProcessed* type events for all scanned objects. If this check box is cleared, Kaspersky Embedded Systems Security does not log *ObjectProcessed* type events. This check box is cleared by default.
Log unprocessed objects	This check box enables or disables the logging *ObjectNotProcessed* type events if a file cannot be processed during a scan. If this check box is selected, Kaspersky Embedded Systems Security logs *ObjectNotProcessed* type events. If this check box is cleared, Kaspersky Embedded Systems Security does not log *ObjectNotProcessed* type events. This check box is cleared by default.
Log packed objects	This check box enables or disables the logging of *PackedObjectDetected* type events for all packed objects that are detected. If this check box is selected, Kaspersky Embedded Systems Security logs *PackedObjectDetected* type events. If this check box is cleared, Kaspersky Embedded Systems Security does not log *PackedObjectDetected* type events. This check box is cleared by default.
Use iChecker technology	This check box enables or disables scan of only new and modified since the last scan files. If the check box is selected, Kaspersky Embedded Systems Security scans only new or modified since the last scan files. If the check box is cleared, Kaspersky Embedded Systems Security scans files regardless to the date of creation or modification. This check box is selected by default.
Use heuristic analysis	This check box enables or disables heuristic analysis during file scans. This check box is selected by default.
Heuristic analysis level	If the **Use heuristic analysis** check box is selected, you can select the heuristic analysis level in the drop-down list: • **Light** is the least detailed scan with minimal system load. • **Medium** is a medium scan with balanced system load. • **Deep** is the most detailed scan with maximum system load. • **Recommended** (default value) is the optimal level recommended by Kaspersky experts. It ensures an optimal combination of protection quality and impact on the performance of the protected devices.

Actions on threat detection window

In this window, you can configure actions to be performed by Kaspersky Embedded Systems Security on detected infected objects:

File Threat Protection settings

Setting	Description
Actions on threat detection	You can select the actions that the application performs on a detected infected object: • **Disinfect; delete if disinfection is not possible** (default). If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. • **Disinfect; block if disinfection fails**. If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application blocks access to the infected object. • **Block**. If you select this action, the application blocks access to the infected object. When an infected object is disinfected or deleted, a copy of the infected object is placed in Backup.

File Threat Protection exclusions

A *protection exclusion* is a set of conditions. When these conditions are met, Kaspersky Embedded Systems Security does not scan the objects for viruses and other malware. You can also exclude objects by masks and threat names, and configure exclusions for processes.

In the Administration Console, you can manage File Threat Protection exclusions in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → File Threat Protection exclusions**).

Settings of scan exclusions

Group of settings	Description
Exclusions	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from scan.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.
Exclusions by threat name	This group of settings contains the **Configure** button, which opens the **Exclusions by threat name** window. In this window, you can configure the exclusion of objects from scans based on threat name.
Exclusions by process	This group of settings contains the **Configure** button, which opens the **Exclusions by process** window. In this window, you can exclude the activity of processes.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

<New exclusion scope> window

In this window, you can add and configure scan exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables exclusion of the scope from scan when the application is running. If this check box is selected, the application excludes this area during scans. If this check box is cleared, the application includes this area in the scan scope. You can later exclude this scope by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	The settings block lets you set the exclusion scope. In the drop-down list of file systems, you can select the type of file system of the directories to be excluded from scans: • **Local**, for local directories. • **Mounted** – mounted directories. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
	If **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the path field. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by threat name window

You can configure the exclusion of objects from scans based on threat name. The application will not block the specified threats. By default, the list of threat names is empty.

You can add, edit, and delete threat names.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by process window

This window contains per-process exclusion scopes. The exclusion scope for exclusion by process lets you exclude from scans the activity of the indicated process and files modified by the indicated process. By default, the list includes two exclusion scopes that contain paths to the Network Agents. You can remove these exclusions, if necessary.

Exclusion scope settings for exclusion by process

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Full path to excluded process.
Status	The status indicates whether the application uses this exclusion.

You can add, edit, or delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

Trusted process window

In this window, you can add and configure exclusion scopes for exclusion by process.

Exclusion scope settings for exclusion by process

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in a table in the **Exclusions by process** window. The entry field must not be blank.
Path to excluded process	Full path to the process you want to exclude from scans.
Apply to child processes	Exclude child processes of the excluded process indicated by the **Path to excluded process** setting. This check box is cleared by default.

Setting	Description
Use this scope	The check box enables or disables this exclusion scope. If this check box is selected, the application excludes this area during scans. If this check box is cleared, the application includes this area in the scan scope. You can later exclude this scope by selecting the check box. This check box is selected by default.
Path to modified files	This group of settings lets you set scan exclusions for files modified by the process. In the drop-down list of file systems, you can select the type of file system of the directories to be excluded from scans: • **Local**, for local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
	If **Mounted** or **Shared** is selected in the drop-down list of file systems, then you can select the remote access protocol in the drop-down list of access protocols: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want add to the exclusion scope. You can use **masks** to specify the path. The entry field must not be blank. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the **Path to modified files** field. You can add, edit, or delete masks.

Configuring File Threat Protection on the command line

In the command line, you can manage File Threat Protection using the File Threat Protection predefined task (*File_Threat_Protection*).

The File Threat Protection task is started by default. You can start and stop this task manually.

Administrator role privileges are required to start and stop the File Threat Protection task from the command line.

You can configure File Threat Protection settings by editing the settings of the File Threat Protection predefined task.

File Threat Protection task settings

The table describes all possible values and default values of all the settings that you can specify for the File Threat Protection task.

File Threat Protection task settings

Setting	Description	Values
`ScanArchived`	Enables scanning of archives (including SFX self-extracting archives). The application scans the following archives: .zip; .7z*; .7-z; .rar; .iso; .cab; .jar; .bz; .bz2; .tbz; .tbz2; .gz; .tgz; .arj. The list of supported archive formats depends on the application databases being used. We do not recommend enabling this setting because scanning archives impacts performance due to the fact that access to the archive file is blocked while the file is being unpacked to a temporary directory and scanned.	`Yes` —Scan archives. If `ActionOnThreat` is set to `DisinfectDeleteIfNotPossible`, and disinfection is impossible, then, depending on the archive type, the application deletes either the infected object or the entire archive that contains the threat. `No` (default value) — Do not scan archives.
`ScanSfxArchived`	Enables scanning of self-extracting archives only (archives that contain an executable extraction module).	`Yes` — Scan self-extracting archives. `No` (default value) — Do not scan self-extracting archives.
`ScanMailBases`	Enables scanning email databases of Microsoft Outlook, Outlook Express, The Bat, and other mail clients.	`Yes` — Scan files of email databases. `No` (default value) — Do not scan files of email databases.
`ScanPlainMail`	Enables scanning of plain text email messages.	`Yes` — Scan plain text email messages. `No` (default value) — Do not scan plain text email messages.
`SkipPlainTextFiles`	Temporary exclusion of files in text format from scans. If the value of this setting is `SkipPlainTextFiles=Yes`, the application does not scan text files if they are reused by the same process for 10 minutes after the most recent scan. This setting makes it possible to optimize scans of application logs.	`Yes` – Do not scan text files if they are reused by the same process for 10 minutes after the most recent scan. `No` (default value) – scan files in plain text format.
`SkipDatabaseJournals`	Temporarily excluding database journal files from scanning. If `SkipDatabaseJournals=yes`, the application does not scan database journal files if the journal file is reused by the same process within 10 minutes after the most recent scan. This setting makes it possible to optimize scans of database journal files.	`Yes` – Do not scan database journal files if they are reused by the same process within 10 minutes after the most recent scan. `No` (default) – Scan database journal files.
`SizeLimit`	Maximum size of an object to be scanned (in megabytes). If the object to be scanned is larger than the specified value, the application skips this object.	0 to 999999 0 — The application scans objects of any size. Default value: 0.
`TimeLimit`	Maximum object scan duration (in seconds). The application stops scanning the object if it takes longer than the time specified by this setting.	0 to 9999 0 — The object scan time is unlimited. Default value: 60.

Setting	Description	Values
`ActionOnThreat`	Selection of the action to be performed by the application on the infected objects.	`DisinfectDeleteIfNotPossible` (disinfect; delete if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. `DisinfectBlockIfNotPossible` (disinfect; block if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application blocks access to the infected object. `Block` – The application blocks access to an infected object. Default value: `DisinfectDeleteIfNotPossible`.
`UseExcludeMasks`	Enables exclusion of the objects specified by the `ExcludeMasks.item_#` setting from the scan.	`Yes` — Exclude objects specified by the `ExcludeMasks.item_#` setting from scans. `No` (default value) — Do not exclude objects specified by the `ExcludeMasks.item_#` setting from the scan.

`ExcludeMasks.item_#`	Excludes objects from being scanned by name or mask.
	You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format.
		The default value is not defined.
		Example:
		`UseExcludeMasks=Yes`
		`ExcludeMasks.item_0000=eicar1.*`
		`ExcludeMasks.item_0001=eicar2.*`

`UseExcludeThreats`	Enables exclusion of objects containing the threats specified by the `ExcludeThreats.item_#` setting from scans.	`Yes` — Exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan. `No` (default value): do not exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan.

`ExcludeThreats.item_#`	Excludes objects from scans by the name of the threats detected in them. Before specifying a value for this setting, make sure that the `UseExcludeThreats` setting is enabled.
	In order to exclude an object from scans, specify the full name of the threat detected in this object – the string containing the application's decision that the object is infected.
	For example, you may be using a utility to obtain information about your network. To keep the application from blocking it, add the full name of the threat contained in it to the list of threats excluded from scans.
	You can find the full name of the threat detected in an object in the application log or on the website https://threats.kaspersky.com.
		The setting value is case-sensitive.
		The default value is not defined.
		Example:
		`UseExcludeThreats=Yes`
		`ExcludeThreats.item_0000=EICAR-Test-*`
		`ExcludeThreats.item_0001=?rojan.Linux`

`ReportCleanObjects`	Enables logging of information about scanned objects that the application reports as not being infected. You can enable this setting, for example, to make sure that a particular object was scanned by the application.	`Yes` — Log information about non-infected objects. `No` (default value) — Do not log information about non-infected objects.
`ReportPackedObjects`	Enables logging of information about scanned objects that are part of compound objects. You can enable this setting, for example, to make sure that an object within an archive has been scanned by the application.	`Yes` — Log information about scanned objects within archives. `No` (default value) — Do not log information about scanned objects within archives.
`ReportUnprocessedObjects`	Enables logging of information about objects that have not been processed for some reason.	`Yes` — Log information about unprocessed objects. `No` (default value) — Do not log information about unprocessed objects.

Setting	Description	Values
`UseAnalyzer`	Enables heuristic analysis. Heuristic analysis helps the application to detect threats even before they become known to virus analysts.	`Yes` (default value) — Enable Heuristic Analyzer. `No` — Disable Heuristic Analyzer.
`HeuristicLevel`	Specifies the heuristic analysis level. The heuristic analysis level sets the balance between the thoroughness of searches for threats, the load on the operating system's resources, and the scan duration. The higher the heuristic analysis level, the more resources and time are required for scanning.	`Light` — The least thorough scan with minimum load on the system. `Medium` — A medium heuristic analysis level with a balanced load on the system. `Deep` — The most thorough scan with maximum load on the system. `Recommended` (default value) — The recommended value.
`UseIChecker`	Enables usage of the iChecker technology.	`Yes` (default value) — Enable use of the iChecker technology. `No` — Disable use of the iChecker technology.
`ScanByAccessType`	File Threat Protection task operation mode. The `ScanByAccessType` setting only applies to the File Threat Protection task.	`SmartCheck` (default value) — Scan a file on attempts to open it, and scan it again on attempts to close it if the file has been modified. If a process accesses an object multiple times in the course of its operation and modifies it, the application scans the object again only when the process closes it for the last time. `OpenAndModify` — Scan a file on attempts to open it, and scan it again on attempts to close it if the file has been modified. `Open` — Scan a file on attempts to open it for reading, execution, or modification.
The **[ScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan scope, which contains additional information about the scan scope. The maximum length of the string specified using this setting is 4096 characters.	Default value: `All objects`. Example: `AreaDesc=" Scanning of email databases "`
`UseScanArea`	Enables scans of the specified scope. To run the task, enable scans of at least one scope.	`Yes` (default value) — Scan the specified scope. `No` — Do not scan the specified scope.
`AreaMask.item_#`	Scan scope limitation. With this scan scope, the application only scans files that are specified using masks in the shell format. If this setting is not specified, the application scans all the objects in the scan scope. You can specify several values for this setting.	The default value is `*` (scan all objects). Example: `AreaMask_item_< item number >=*doc`

Setting	Description	Values
`Path`	Path to the directory with objects to be scanned.	`< path to local directory >` — scan objects in the specified directory. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. `Shared:NFS` — Scan the device file system resources that are accessible via the NFS protocol. `Shared:SMB` – Scan the device file system resources that are accessible via the Samba protocol. `Mounted:NFS` – Scan the remote directories mounted on a device using the NFS protocol. `Mounted:SMB` – Scan the remote directories mounted on a device using the Samba protocol. `AllRemoteMounted` – Scan all remote directories mounted on the device using the Samba and NFS protocols. `AllShared` – Scan all the device file system resources that are accessible via the Samba and NFS protocols. `<file system type>` – Scan all resources of the specified file system device.
The **[ExcludedFromScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan exclusion scope, which contains additional information about the exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from scans.	`Yes` (default value) — Exclude the specified scope. `No` — Do not exclude the specified scope.
`AreaMask.item_#`	Limitation of scan exclusion scope. In the exclusion scope, the application excludes from scans only files that are specified using masks in the shell format. If this setting is not specified, the application does not scan any of the objects within the exclusion scope. You can specify several values for this setting.	Default value: `*` (exclude all objects from scan)

Setting	Description	Values
Path	Path to the directory with objects to be excluded.	< path to local directory > – exclude objects in the specified directory (including subdirectories) from scans. You can use **masks** to specify the path. You can use the * (asterisk) character to create a file or directory name mask. You can indicate a single * character to represent any set of characters (including an empty set) preceding the / character in the file or directory name. For example, /dir/*/file or /dir/*/*/file. You can indicate two consecutive * characters to represent any set of characters (including an empty set and the / character) in the file or directory name. For example, /dir/**/file*/ or /dir/file**/. The ** mask can be used only once in a directory name. For example, /dir/**/**/file is an incorrect mask. You can use a single ? character to represent any one character in the file or directory name. Mounted:NFS – Exclude the remote directories mounted on a device using the NFS protocol from scan. Mounted:SMB – Exclude the remote directories mounted on a device using the Samba protocol from scan. AllRemoteMounted – Exclude all remote directories mounted on the device using the Samba and NFS protocols from scan. <file system type> — Exclude all the resources of the specified device file system from scans.
The **[ExcludedForProgram.item_#]** section contains the following settings:		
ProgramPath	Path to excluded process.	< full path to process > – Do not scan the process in the specified local directory.
ApplyToDescendants	Exclude child processes of the excluded process specified by the ProgramPath setting from scans.	Yes – exclude the specified process and all its child processes from scans. No (default value) – exclude only the specified process from scans, do not exclude its child processes from scans.
AreaDesc	Description of the process exclusion scope.	Default value: All objects.
UseExcludedForProgram	Excludes the specified scope from scans.	Yes (default value) — Exclude the specified scope. No — Do not exclude the specified scope.
AreaMask.item_#	Limitation of the process exclusion scope. In the process exclusion scope, the application excludes from scans only the files that are specified using masks in the shell format. If this setting is not specified, the application excludes from scans all the objects within the process exclusion scope. You can specify several values for this setting.	Default value: * (exclude all objects from scan)

Setting	Description	Values
`Path`	Path to a directory with files that are modified by the process.	`< path to local directory >` excludes objects in the specified directory from the scan. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. `Shared:NFS` — Exclude device file system resources that are accessible via the NFS protocol from scans. `Shared:SMB` — Exclude device file system resources that are accessible via the Samba protocol from scans. `Mounted:NFS` – Exclude the remote directories mounted on a device using the NFS protocol from scan. `Mounted:SMB` – Exclude the remote directories mounted on a device using the Samba protocol from scan. `AllRemoteMounted` – Exclude all remote directories mounted on the device using the Samba and NFS protocols from scan. `AllShared` – Exclude all device file system resources that are accessible using the Samba and NFS protocols from scan. `< file system type >` – Exclude all resources of the specified device file system from the scan.

Optimizing network directory scanning

To optimize the File Threat Protection task, you can exclude from scans any files being copied from network directories to the local directory. To do so, configure exclusion based on processes for the utility used for copying from network directories (for example, for the `cp` utility).

To configure exclusion of network directories from scans:

1. Output the File Threat Protection task settings (*File_Threat_Protection*, ID:1) to a configuration file using the following command:

   ```
   kess-control --get-settings 1 --file < full path to configuration file > [--json]
   ```

2. Open the configuration file and add the `[ExcludedForProgram.item_#]` section with the following settings:

   - `ProgramPath` – path to the process to be excluded or to the directory with the processes to be excluded.
   - `ApplyToDescendants` is a parameter that indicates whether the scan should exclude child processes of the excluded process (possible values: `Yes` or `No`).
   - `AreaDesc` – a description of the process exclusion scope, which contains additional information about the exclusion scope.

- `UseExcludedForProgram` enables exclusion of the specified scope during task operation (possible values: `Yes` or `No`).
- `Path` – path to the files or directory with files modified by the process.
- `AreaMask.item_#` is the file name mask for the files to be excluded from the scan. You can also specify the full path to the file.

Example:

```
[ExcludedForProgram.item_0000]
ProgramPath=/usr/bin/cp
ApplyToDescendants=No
AreaDesc=
UseExcludedForProgram=Yes
Path=AllRemoteMounted
AreaMask.item_0000=*
```

3. Execute the command:

   `kess-control --set-settings 1 --file <full path to configuration file> [--json]`

   Specify the `--json` option if you are importing settings from a configuration file in JSON format. If the option is not specified, the application will attempt to import settings from an INI file. If the import fails, an error is displayed.

The application does not scan the files in network directories, but the `cp` command itself (for the example given above) and local files are scanned.

Special considerations for scanning symbolic links and hard links

Kaspersky Embedded Systems Security lets you scan symbolic links and hard links to files.

Scanning symbolic links

The application scans symbolic links only if the file referenced by the symbolic link is within the scan scope of the File Threat Protection component.

If the file referenced by the symbolic link is not within the scan scope of the File Threat Protection component, the application does not scan this file. However, if the file contains malicious code, the security of the device is at risk.

Scanning hard links

When processing a file that has more than one hard link, the application disinfects the original file. If disinfection fails, the application deletes the hard link and creates in its place a copy of the source file with the name of the deleted hard link. When deleting, the application deletes the hard link being processed, but does not process the other hard links to this file.

When you restore a file with a hard link from the Backup storage, the application creates a copy of the source file with the name of the hard link that was moved to the Backup storage. Connections with the remaining hard links to the source file will not be restored.

Malware Scan

Malware Scan is a one-time full or custom file scan on the device performed on demand. Kaspersky Embedded Systems Security can carry out multiple Malware Scan tasks at the same time.

A *Malware Scan (Scan_My_Computer)* predefined task is created in the application. You can use this task to perform a full scan of the device. During a full scan, the application scans all objects located on the device's local drives, as well as all mounted and shared objects that are accessed via Samba or NFS protocols with the recommended security settings.

If the root user does not have access to a remote directory mounted on the device over the NFSv4 protocol, the application blocks access to such a file for all users.

In Kaspersky Security Center, the Kaspersky Security Center Initial Configuration Wizard automatically creates a Malware Scan group task after installing the administration MMC plug-in or the Kaspersky Embedded Systems Security administration web plug-in.

During a full disk scan, the processor is busy. It is recommended to run the full scan task when the business is idle.

You can configure the settings of automatically created tasks in Kaspersky Security Center and on the command line, and also create Malware Scan user tasks.

Upon detecting malware, Kaspersky Embedded Systems Security may remove the infected file and terminate the malware process started from this file.

If during execution of the malware scan task the application was restarted by a control service or manually by the user, the task will be stopped. The application logs the *OnDemandTaskInterrupted* event.

You can run Malware Scan tasks and configure scan settings:

- Select operating system objects to scan: files, archives, boot sectors, process memory and kernel memory, startup objects.
- Limit the size of an object to be scanned and the duration of the object scan.
- Select the actions to be performed by the application on the infected objects.
- Configure exclusions of objects from scans:
  - by name or mask
  - by the name of the threats detected in the objects

  When running on operating systems that use the special-purpose parsecfs file system (Astra Linux Special Edition) we recommend adding mounting paths of this file system to exclusions from scanning.
- Enable or disable global exclusions and File Threat Protection exclusions when scanning.

- Enable the logging of information about scanned non-infected objects, about scanning objects in archives, and about unprocessed objects.
- Configure the use of the heuristic analyzer and iChecker technology during a scan.
- Limit the set of devices whose boot sectors need to be scanned.
- Configure scan scopes and scan exclusion scopes.

Malware Scan in the Web Console

In the Web Console, you can scan for malware using the *Malware Scan* task.

You can run an automatically created group task and also create and run user tasks for scanning. You can configure scan settings by editing the settings of Malware Scan tasks.

Malware Scan task settings

Setting	Description
Scan archives	This check box enables or disables scan of archives. If the check box is selected, the application scans the archives. To scan an archive, the application has to unpack it first, which may slow down scanning. You can reduce the duration of archive scans by configuring the **Skip file that is scanned for longer than (sec)** and **Skip file larger than (MB)** settings in the **General scan settings** section. If the check box is cleared, the application does not scan the archives. This check box is selected by default.
Scan SFX archives	This check box enables or disables *self-extracting archive* scans. Self-extracting archives are the archives that contain an executable extraction module. If the check box is selected, the application scans self-extracting archives. If the check box is cleared, the application does not scan self-extracting archives. This check box is available if the **Scan archives** check box is unchecked. This check box is selected by default.
Scan mail databases	This check box enables or disables scans of mail databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail applications. If the check box is selected, the application scans mail database files. If the check box is cleared, the application does not scan mail database files. This check box is cleared by default.
Scan mail format files	This check box enables or disables scan of files of plain-text email messages. If this check box is selected, the application scans plain-text messages. If this check box is cleared, the application does not scan plain-text messages. This check box is cleared by default.
Skip file that is scanned for longer than (sec)	In this field, you can specify the maximum time to scan a file, in seconds. After the specified time, the application stops scanning the file. Possible values: 0–9999. If the value is set to 0, the scan time is unlimited. Default value: 0.
Skip file larger than (MB)	In this field, you can specify the maximum size of a file to scan, in megabytes. Possible values: 0–999999. If the value is set to 0, the application scans files of any size. Default value: 0.
Log clean objects	This check box enables or disables the logging of *ObjectProcessed* type events. If this check box is selected, the application logs events of the *ObjectProcessed* type for all scanned objects. If this check box is cleared, the application does not log events of the *ObjectProcessed* type for any scanned object. This check box is cleared by default.

Setting	Description
Log unprocessed objects	This check box enables or disables the logging *ObjectNotProcessed* type events if a file cannot be processed during a scan. If this check box is selected, the application logs the events of the *ObjectNotProcessed* type. If this check box is cleared, the application does not log the events of the *ObjectNotProcessed* type. This check box is cleared by default.
Log packed objects	This check box enables or disables the logging of *PackedObjectDetected* type events for all packed objects that are detected. If this check box is selected, the application logs the events of the *PackedObjectDetected* type. If this check box is cleared, the application does not log the events of the *PackedObjectDetected* type. This check box is cleared by default.
Use iChecker technology	This check box enables or disables scan of only new and modified since the last scan files. If the check box is selected, the application scans only new files or the files modified since the last scan. If the check box is cleared, the application scans the files regardless of the creation or modification date. This check box is selected by default.
Use heuristic analysis	This check box enables or disables heuristic analysis during file scans. This check box is selected by default.
Heuristic analysis level	If the **Use heuristic analysis** check box is selected, you can select the heuristic analysis level in the drop-down list: • **Light** is the least detailed scan with minimal system load. • **Medium** is a medium scan with balanced system load. • **Deep** is the most detailed scan with maximum system load. • **Recommended** (default value) is the optimal level recommended by Kaspersky experts. It ensures an optimal combination of protection quality and impact on the performance of the protected devices.
Actions on threat detection	You can select the action that the application performs on a detected infected object: • **Disinfect; delete if disinfection is not possible** (default). If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. • **Disinfect; inform if disinfection fails**. If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. • **Inform**. If you select this action, the application creates a record about the infected object in the log.
Scan scopes	The table that contains the scopes scanned by the task. By default, the table contains one scan scope that includes all directories of the local file system. You can add, modify, delete, move up, or move down scan scopes in the table. You can also import the list of scan scopes from a file by clicking **Import** or export the list of scan scopes to a file in JSON format by clicking **Export**. When importing, you are prompted to replace the list of scan scopes or add scan scopes to the existing list.

Add scan scope window

In this window, you can add and configure scan scopes.

Scan scope settings

Setting	Description
Scope name	Field for entering the scan scope name. This name is displayed in the **Scan scopes** table in the **Scan settings** section. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this scan scope. If this check box is cleared, the application does not process this scan scope. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	You can select the type of file system in the drop-down list: • **Local** (default value) – local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
Access protocol	You can select the remote access protocol in the drop-down list: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below. This drop-down list is available if the **Shared** or **Mounted** type is selected in the drop-down list of file systems.
Path	This is the entry field for specifying the path to the directory that you want to include in the scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default – the application scans all directories of the local file system. This field is available if the **Local** type is selected in the drop-down list of file systems. If the **Local** type is selected in the drop-down list of file systems, and the path is not specified, the application scans all directories of the local file system.
Name of shared resource	The field for entering the name of the file system shared resource, where the directories that you want to add to the scan scope are located. The field is available if the **Mounted** type is selected in the File system drop-down list and the **Custom** item is selected in the **Access protocol** drop-down list.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Scan scopes section

You can configure scan scope settings for the Malware Scan task. The application allows you to scan files, boot sectors, client device memory, and startup objects.

Malware Scan scope task settings

Setting	Description
Scan files	This check box enables or disables file scans. If the check box is selected, the application scans the files. If the check box is cleared, the application does not scan the files. This check box is selected by default.
Scan boot sectors	This check box enables or disables boot sector scans. If the check box is selected, the application scans the boot sectors. If the check box is cleared, the application does not scan the boot sectors. This check box is cleared by default.

Setting	Description
Scan kernel memory and running processes	This check box enables or disables client device memory scan. If the check box is selected, the application scans kernel memory and running processes. If the check box is cleared, the application does not scan kernel memory and running processes. This check box is cleared by default.
Scan startup objects	This check box enables or disables startup object scans. If the check box is selected, the application scans startup objects. If the check box is cleared, the application does not scan startup objects. This check box is cleared by default.
Devices to scan	Clicking the **Configure device masks** link opens the **Scan scopes** window, where you can specify the devices whose boot sectors will be scanned.

Scan scopes window

The table contains name masks of the devices, whose boot sectors the application must scan. By default, the table contains the **/**** device name mask (all devices).

You can add, edit, and delete items in the table.

Exclusion scopes section

In the **Exclusion scopes** section for the Malware Scan task, you can configure exclusion scopes, exclusions by mask and threat name, as well as the use of global exclusions and File Threat Protection exclusions when the task is running.

Settings of scan exclusions

Setting	Description
Configure exclusion scopes	Clicking the **Configure exclusion scopes** link opens the **Exclusion scopes** window. In this window, you can define the list of scan exclusions.
Configure exclusions by mask	Clicking the **Configure exclusions by mask** link opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.
Configure exclusions by threat name	Clicking the **Configure exclusions by threat name** link opens the **Exclusions by threat name** window. In this window, you can configure the exclusion of objects from scans based on threat name.
Use global exclusions	The check box enables or disables the exclusion of the mount points specified in global exceptions while the application is running. If this check box is selected, the application excludes configured mount points from scans. This check box is selected by default.
Use File Threat Protection exclusions	This check box enables or disables the use of configured File Threat Protection exclusions when the application is running. If the check box is selected, the application does not scan the objects specified in the exclusions for the File Threat Protection component. This check box is selected by default.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

Add exclusion scope window

In this window, you can add and configure exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables the exclusion of the scope when the application is running. If the check box is selected, the application excludes this scope from scan or protection during its operation. If the check box is cleared, the application includes this scope in scan or protection during its operation. You can later exclude this scope from scan or protection by selecting the check box. This check box is selected by default.
File system, access protocol, and path	In this drop-down list, you can select the type of file system where the directories that you want to add to scan exclusions are located: • **Local**, for local directories. • **Mounted**, for remote directories mounted on the device. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
Access protocol	You can select the remote access protocol in the drop-down list: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below. This drop-down list is available if the **Mounted** type is selected in the drop-down list of file systems.

Setting	Description
Path	Entry field for the path to the directory that you want to add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan. This field is available if the **Local** type is selected in the drop-down list of file systems.
Name of shared resource	The field for entering the name of the file system shared resource, where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the File system drop-down list and the **Custom** item is selected in the **Access protocol** drop-down list.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the **Path** field. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** or **Edit** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by threat name window

You can configure the exclusion of objects from scans based on threat name. The application will not block the specified threats. By default, the list of threat names is empty.

You can add, edit, and delete threat names.

Clicking the **Add** or **Edit** button opens the **Threat name** window. In this window, you can define the name of the threat to be excluded from a scan.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Malware Scan in the Administration Console

In the Administration Console, you can scan for malware using the *Malware Scan* task.

You can run an automatically created group task and also create and run user tasks for scanning. You can configure scan settings by editing the settings of Malware Scan tasks.

In the **Settings** section of the properties of the Malware Scan task, you can configure the settings listed in the table below.

Malware Scan task settings

Setting	Description
Scan	This group of settings contains buttons that open windows where you can configure the scan scopes, scan scope settings, and scan settings.
Actions on threat detection	This group of settings contains the **Configure** button. Clicking this button opens the **Actions on threat detection** window, where you can configure the actions that the application performs on detected infected objects.

In the **Exclusions** section, you can configure exclusion scopes as well as exclusions by mask and by threat name in the properties of the Malware Scan task.

Scan scopes window

The table contains the scan scopes. The application will scan files and directories located in the paths specified in the table. By default, the table contains one scan scope that includes all directories of the local file system.

Scan scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Scope name	Scan scope name.
Path	Path to the directory that the application scans.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure scan scopes.

Scan scope settings

Setting	Description
Scan scope name	Field for entering the scan scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this scan scope. If this check box is cleared, the application does not process this scan scope. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	The settings block lets you set the scan scope. You can select the file system type in the drop-down list of file systems: • **Local** (default value) – local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
	If **Shared** or **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want to add to the scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default – the application scans all directories of the local file system. If the **Local** type is selected in the drop-down list of file systems, and the path is not specified, the application scans all directories of the local file system.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the scan scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Scan scope settings window

In this window, you can configure the scan settings for the Malware Scan task. The application allows you to scan files, boot sectors, device memory, and startup objects.

Scan scope settings

Setting	Description
Scan files	This check box enables or disables file scans. If the check box is selected, the application scans the files. If the check box is cleared, the application does not scan the files. This check box is selected by default.
Scan boot sectors	This check box enables or disables boot sector scans. If the check box is selected, the application scans the boot sectors. If the check box is cleared, the application does not scan the boot sectors. This check box is cleared by default.

Setting	Description
Scan kernel memory and running processes	This check box enables or disables device memory scan. If the check box is selected, the application scans kernel memory and running processes. If the check box is cleared, the application does not scan kernel memory and running processes. This check box is cleared by default.
Scan startup objects	This check box enables or disables startup object scans. If the check box is selected, the application scans startup objects. If the check box is cleared, the application does not scan startup objects. This check box is cleared by default.
Devices to scan	This group of settings contains the **Configure** button. Clicking this button opens the **Scan scopes** window, where you can specify the devices whose boot sectors must be scanned.
Use global exclusions	The check box enables or disables the exclusion of the mount points specified in global exceptions while the application is running. If this check box is selected, the application excludes configured mount points from scans. This check box is selected by default.
Use File Threat Protection exclusions	This check box enables or disables the use of configured File Threat Protection exclusions when the application is running. If the check box is selected, the application does not scan the objects specified in the exclusions for the File Threat Protection component. This check box is selected by default.

Scan scopes window

The table contains name masks of the devices, whose boot sectors the application must scan. By default, the table contains the **/**** device name mask (all devices).

You can add, edit, and delete items in the table.

Scan settings window

In this window, you can configure the file scan settings for the task.

Scan settings

Setting	Description
Scan archives	This check box enables or disables scan of archives. If the check box is selected, the application scans the archives. To scan an archive, the application has to unpack it first, which may slow down scanning. You can reduce the duration of archive scans by configuring the **Skip file that is scanned for longer than (sec)** and **Skip file larger than (MB)** settings in the **General scan settings** section. If the check box is cleared, the application does not scan the archives. This check box is selected by default.
Scan SFX archives	This check box enables or disables *self-extracting archive* scans. Self-extracting archives are the archives that contain an executable extraction module. If the check box is selected, the application scans self-extracting archives. If the check box is cleared, the application does not scan self-extracting archives. This check box is available if the **Scan archives** check box is unchecked. This check box is selected by default.

Setting	Description
Scan mail databases	This check box enables or disables scans of mail databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail applications. If the check box is selected, the application scans mail database files. If the check box is cleared, the application does not scan mail database files. This check box is cleared by default.
Scan mail format files	This check box enables or disables scan of files of plain-text email messages. If this check box is selected, the application scans plain-text messages. If this check box is cleared, the application does not scan plain-text messages. This check box is cleared by default.
Skip file that is scanned for longer than (sec)	In this field, you can specify the maximum time to scan a file, in seconds. After the specified time, the application stops scanning the file. Possible values: 0–9999. If the value is set to 0, the scan time is unlimited. Default value: 0.
Skip file larger than (MB)	In this field, you can specify the maximum size of a file to scan, in megabytes. Possible values: 0–999999. If the value is set to 0, the application scans files of any size. Default value: 0.
Log clean objects	This check box enables or disables the logging of *ObjectProcessed* type events. If this check box is selected, the application logs events of the *ObjectProcessed* type for all scanned objects. If this check box is cleared, the application does not log events of the *ObjectProcessed* type for any scanned object. This check box is cleared by default.
Log unprocessed objects	This check box enables or disables the logging *ObjectNotProcessed* type events if a file cannot be processed during a scan. If this check box is selected, the application logs the events of the *ObjectNotProcessed* type. If this check box is cleared, the application does not log the events of the *ObjectNotProcessed* type. This check box is cleared by default.
Log packed objects	This check box enables or disables the logging of *PackedObjectDetected* type events for all packed objects that are detected. If this check box is selected, the application logs the events of the *PackedObjectDetected* type. If this check box is cleared, the application does not log the events of the *PackedObjectDetected* type. This check box is cleared by default.
Use iChecker technology	This check box enables or disables scan of only new and modified since the last scan files. If the check box is selected, the application scans only new files or the files modified since the last scan. If the check box is cleared, the application scans the files regardless of the creation or modification date. This check box is selected by default.
Use heuristic analysis	This check box enables or disables heuristic analysis during file scans. This check box is selected by default.
Heuristic analysis level	If the **Use heuristic analysis** check box is selected, you can select the heuristic analysis level in the drop-down list: • **Light** is the least detailed scan with minimal system load. • **Medium** is a medium scan with balanced system load. • **Deep** is the most detailed scan with maximum system load. • **Recommended** (default value) is the optimal level recommended by Kaspersky experts. It ensures an optimal combination of protection quality and impact on the performance of the protected devices.

Actions on threat detection window

In this window, you can configure actions to be performed by Kaspersky Embedded Systems Security on detected infected objects:

Actions on threat detection

Setting	Description
Actions on threat detection	You can select the action that the application performs on a detected infected object: • **Disinfect; delete if disinfection is not possible** (default). If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. • **Disinfect; inform if disinfection fails**. If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. • **Inform**. If you select this action, the application creates a record about the infected object in the log.

Exclusions section

Scan exclusion is a set of conditions. When these conditions are met, Kaspersky Embedded Systems Security does not scan the objects for viruses and other malware. You can also exclude objects from scans by masks and threat names.

Settings of scan exclusions

Group of settings	Description
Exclusion scopes	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from scans.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.
Exclusions by threat name	This group of settings contains the **Configure** button, which opens the **Exclusions by threat name** window. In this window, you can configure the exclusion of objects from scans based on threat name.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

<New exclusion scope> window

In this window, you can add and configure scan exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables exclusion of the scope from scan when the application is running. If this check box is selected, the application excludes this area during scans. If this check box is cleared, the application includes this area in the scan scope. You can later exclude this scope by selecting the check box. This check box is selected by default.
File system, access protocol, and path	The settings block lets you set the exclusion scope. In the drop-down list of file systems, you can select the type of file system of the directories to be excluded from scans: • **Local**, for local directories. • **Mounted** – mounted directories. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
	If **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the path field. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by threat name window

You can configure the exclusion of objects from scans based on threat name. The application will not block the specified threats. By default, the list of threat names is empty.

You can add, edit, and delete threat names.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Malware Scan in the command line

On the command line, you can scan for malware in the following ways:

- Using the Malware Scan predefined task (*Scan_My_Computer*). You can manually start, stop, pause, or resume this task and configure the task run schedule. You can configure scan settings by editing the settings of this task.
- Using Malware Scan user tasks (tasks of the *ODS* type). You can manually start, stop, pause, or resume user tasks and configure the task schedule.

Using the `kess-control --scan-file` command, you can perform a *custom scan* of the specified files and directories.

Settings of the Malware Scan predefined task

The table describes all possible values and the default values of all the settings that you can specify for the Malware Scan task.

Malware Scan task settings

Setting	Description	Values
`ScanFiles`	Enables file scan.	`Yes` (default value) — Scan files. `No` — Do not scan files.
`ScanBootSectors`	Enables boot sector scans.	`Yes` (default value) — Scan boot sectors. `No` — Do not scan boot sectors.
`ScanComputerMemory`	Enables process memory and kernel memory scans.	`Yes` (default value) — Scan process memory and kernel memory. `No` — Do not scan process memory and kernel memory.
`ScanStartupObjects`	Enables startup object scans.	`Yes` (default value) — Scan startup objects. `No` — Do not scan startup objects.
`ScanArchived`	Enables scanning of archives (including SFX self-extracting archives). The application scans the following archives: .zip; .7z*; .7-z; .rar; .iso; .cab; .jar; .bz; .bz2; .tbz; .tbz2; .gz; .tgz; .arj. The list of supported archive formats depends on the application databases being used.	`Yes` (default value) — Scan archives. If `ActionOnThreat` is set to `DisinfectDeleteIfNotPossible`, and disinfection is impossible, then, depending on the archive type, the application deletes either the infected object or the entire archive that contains the threat. `No` — Do not scan archives.
`ScanSfxArchived`	Enables scanning of self-extracting archives only (archives that contain an executable extraction module).	`Yes` (default value) — Scan self-extracting archives. `No` — Do not scan self-extracting archives.
`ScanMailBases`	Enables scanning email databases of Microsoft Outlook, Outlook Express, The Bat, and other mail clients.	`Yes` — Scan files of email databases. `No` (default value) — Do not scan files of email databases.
`ScanPlainMail`	Enables scanning of plain text email messages.	`Yes` — Scan plain text email messages. `No` (default value) — Do not scan plain text email messages.
`SizeLimit`	Maximum size of an object to be scanned (in megabytes). If the object to be scanned is larger than the specified value, the application skips this object.	0 to 999999 0 — The application scans objects of any size. Default value: 0.
`TimeLimit`	Maximum object scan duration (in seconds). The application stops scanning the object if it takes longer than the time specified by this setting.	0 to 9999 0 — The object scan time is unlimited. Default value: 0.
`ActionOnThreat`	Selection of the action to be performed by the application on the infected objects.	`DisinfectDeleteIfNotPossible` (disinfect; delete if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. `DisinfectInformIfNotPossible` (disinfect; inform if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. `Inform` means the application does not try to disinfect or delete the infected object. Information about the infected object is logged. Default value: `DisinfectDeleteIfNotPossible`.

Setting	Description	Values
`UseExcludeMasks`	Enables exclusion of the objects specified by the `ExcludeMasks.item_#` setting from the scan.	`Yes` — Exclude objects specified by the `ExcludeMasks.item_#` setting from scans. `No` (default value) — Do not exclude objects specified by the `ExcludeMasks.item_#` setting from the scan.

`ExcludeMasks.item_#`	Excludes objects from being scanned by name or mask. You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format.
	Before specifying a value for this setting, make sure that the `UseExcludeMasks` setting is enabled.
		The default value is not defined.
		Example:
		`UseExcludeMasks=Yes`
		`ExcludeMasks.item_0000=eicar1.*`
		`ExcludeMasks.item_0001=eicar2.*`

`UseExcludeThreats`	Enables exclusion of objects containing the threats specified by the `ExcludeThreats.item_#` setting from scans.	`Yes` — Exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan. `No` (default value): do not exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan.

`ExcludeThreats.item_#`	Excludes objects from scans by the name of the threats detected in them. Before specifying a value for this setting, make sure that the `UseExcludeThreats` setting is enabled.
	In order to exclude an object from scans, specify the full name of the threat detected in this object – the string containing the application's decision that the object is infected.
	For example, you may be using a utility to obtain information about your network. To keep the application from blocking it, add the full name of the threat contained in it to the list of threats excluded from scans.
	You can find the full name of the threat detected in an object in the application log or on the website https://threats.kaspersky.com.
		The setting value is case-sensitive.
		The default value is not defined.
		Example:
		`UseExcludeThreats=Yes`
		`ExcludeThreats.item_0000=EICAR-Test-*`
		`ExcludeThreats.item_0001=?rojan.Linux`

`UseGlobalExclusions`	Enables global exclusions for scanning.	`Yes` (default value) — Use the global exclusions. `No` — Do not use global exclusions.
`UseOASExclusions`	Enables File Threat Protection exclusions for scanning.	`Yes` (default value) — Use File Threat Protection exclusions. `No` — do not use File Threat Protection exclusions.
`ReportCleanObjects`	Enables logging of information about scanned objects that the application reports as not being infected. You can enable this setting, for example, to make sure that a particular object was scanned by the application.	`Yes` — Log information about non-infected objects. `No` (default value) — Do not log information about non-infected objects.
`ReportPackedObjects`	Enables logging of information about scanned objects that are part of compound objects. You can enable this setting, for example, to make sure that an object within an archive has been scanned by the application.	`Yes` — Log information about scanned objects within archives. `No` (default value) — Do not log information about scanned objects within archives.
`ReportUnprocessedObjects`	Enables logging of information about objects that have not been processed for some reason.	`Yes` — Log information about unprocessed objects. `No` (default value) — Do not log information about unprocessed objects.
`UseAnalyzer`	Enables heuristic analysis. Heuristic analysis helps the application to detect threats even before they become known to virus analysts.	`Yes` (default value) — Enable Heuristic Analyzer. `No` — Disable Heuristic Analyzer.

Setting	Description	Values
`HeuristicLevel`	Specifies the heuristic analysis level. You can specify the heuristic analysis level. The heuristic analysis level sets the balance between the thoroughness of searches for threats, the load on the operating system's resources, and the scan duration. The higher the heuristic analysis level, the more resources and time are required for scanning.	`Light` — The least thorough scan with minimum load on the system. `Medium` — A medium heuristic analysis level with a balanced load on the system. `Deep` — The most thorough scan with maximum load on the system. `Recommended` (default value) — The recommended value.
`UseIChecker`	Enables usage of the iChecker technology.	`Yes` (default value) — Enable use of the iChecker technology. `No` — Disable use of the iChecker technology.
`DeviceNameMasks.item_#`	List of device names. The application will scan boot sectors of these devices. The setting value cannot be empty. At least one device name mask must be specified to run this task.	`AllObjects` – scan boot sectors of all devices. `< device name mask >` – scan boot sectors of the devices whose names match the specified mask. Default value: `/**` – any set of characters in the device name, including the `/` character.
The **[ScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan scope, which contains additional information about the scan scope. The maximum length of the string specified using this setting is 4096 characters.	Default value: `All objects`. Example: `AreaDesc=" Mail bases scan "`
`UseScanArea`	Enables scans of the specified scope. To run the task, enable scans of at least one scope.	`Yes` (default value) — Scan the specified scope. `No` — Do not scan the specified scope.
`AreaMask.item_#`	Scan scope limitation. Within the scan scope, the application scans only the files that are specified using the masks in the shell format. If this setting is not specified, the application scans all the objects in the scan scope. You can specify several values for this setting.	The default value is `*` (scan all objects). Example: `AreaMask.item_< item number >=*doc`
`Path`	Path to the directory with objects to be scanned.	`< path to local directory >` — scan objects in the specified directory. `Shared:NFS` — Scan the device file system resources that are accessible via the NFS protocol. `Shared:SMB` – Scan the device file system resources that are accessible via the Samba protocol. `Mounted:NFS` – Scan the remote directories mounted on a device using the NFS protocol. `Mounted:SMB` – Scan the remote directories mounted on a device using the Samba protocol. `AllRemoteMounted` – Scan all remote directories mounted on the device using the Samba and NFS protocols. `AllShared` – Scan all the device file system resources that are accessible via the Samba and NFS protocols. `<file system type>` – Scan all resources of the specified file system device.
The **[ExcludedFromScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan exclusion scope, which contains additional information about the exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from scans.	`Yes` (default value) — Exclude the specified scope. `No` — Do not exclude the specified scope.

Setting	Description	Values
`AreaMask.item_#`	Limitation of scan exclusion scope. In the exclusion scope, the application excludes only the files that are specified using masks in the shell format. If this setting is not specified, the application excludes all the objects in the exclusion scope. You can specify several values for this setting.	Default value: `*` (exclude all objects)
`Path`	Path to the directory with objects to be excluded.	`<path to local directory>` – Exclude objects in the specified directory (including subdirectories) from scans. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. In order to optimize the operation of scan tasks, it is recommended to add the path with snapshots mounted by the system in the read-only mode to the exclusions for the systems with the btrfs file system and enabled active snapshots. For example, for the systems based on SUSE/OpenSUSE, you can add the following exclusion `/.snapshots/*/snapshot/`. The `/` path is specified by default. The application excludes all directories of the local file system from scan. `Mounted:NFS` – Exclude the remote directories mounted on a device using the NFS protocol from scan. `Mounted:SMB` – Exclude the remote directories mounted on a device using the Samba protocol from scan. `AllRemoteMounted` – Exclude all remote directories mounted on the device using the Samba and NFS protocols from scan. `<file system type>` – Exclude from scanning all resources of the specified file system device. Remote directories are excluded from scanning by the application only if they were mounted before the task was started. Remote directories mounted after the task is started are not excluded from scanning.

Custom Scan of files and directories on the command line

You can perform a custom scan of the specified files and directories using the following command: `kess-control --scan-file`.

A custom scan is performed with the settings stored in the *Scan_File* predefined task (ID:3). You can configure settings for a custom scan of files by editing the settings of this task (see the table below).

To start a custom scan of the specified files and directories, execute the following command:

```
kess-control --scan-file <path> [--action <action>]
```

where:

- `<path>` is the path to the file or directory that you want to scan. You can specify multiple paths by separating them with a space.

- `--action <action>` is the action to be performed by the application on the infected objects. Possible values: `DisinfectDeleteIfNotPossible, DisinfectInformIfNotPossible, Inform`. If you do not specify the `--action` option, the application performs the `DisinfectDeleteIfNotPossible` action.

As a result of executing the command, a temporary file scan task is created, which is automatically deleted after completion. In this case, the scan results are output to the console.

The table describes all possible values and the default values of all the settings that you can specify for the *Scan_File* task.

The `[ScanScope.item_#]` and `[ExcludedFromScanScope.item_#]` sections defined in the *Scan_File* task are not taken into account when performing the custom scan.

Scan_File task settings

Setting	Description	Values
`ScanFiles`	Enables file scan.	`Yes` (default value) — Scan files. `No` — Do not scan files.
`ScanBootSectors`	Enables boot sector scans.	`Yes` — Scan boot sectors. `No` (default value) — Do not scan boot sectors.
`ScanComputerMemory`	Enables process memory and kernel memory scans.	`Yes` — Scan process memory and kernel memory. `No` (default value) — Do not scan process memory and kernel memory.
`ScanStartupObjects`	Enables startup object scans.	`Yes` — Scan startup objects. `No` (default value) — Do not scan startup objects.
`ScanArchived`	Enables scanning of archives (including SFX self-extracting archives). The application scans the following archives: .zip; .7z*; .7-z; .rar; .iso; .cab; .jar; .bz; .bz2; .tbz; .tbz2; .gz; .tgz; .arj. The list of supported archive formats depends on the application databases being used.	`Yes` (default value) — Scan archives. If `ActionOnThreat` is set to `DisinfectDeleteIfNotPossible`, and disinfection is impossible, then, depending on the archive type, the application deletes either the infected object or the entire archive that contains the threat. `No` — Do not scan archives.
`ScanSfxArchived`	Enables scanning of self-extracting archives only (archives that contain an executable extraction module).	`Yes` (default value) — Scan self-extracting archives. `No` — Do not scan self-extracting archives.
`ScanMailBases`	Enables scanning email databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail clients.	`Yes` — Scan files of email databases. `No` (default value) — Do not scan files of email databases.
`ScanPlainMail`	Enables scanning of plain text email messages.	`Yes` — Scan plain text email messages. `No` (default value) — Do not scan plain text email messages.
`SizeLimit`	Maximum size of an object to be scanned (in megabytes). If the object to be scanned is larger than the specified value, the application skips this object.	0 to 999999 0 — The application scans objects of any size. Default value: 0.

Setting	Description	Values
`TimeLimit`	Maximum object scan duration (in seconds). The application stops scanning the object if it takes longer than the time specified by this setting.	0 to 9999 0 — The object scan time is unlimited. Default value: 0.
`ActionOnThreat`	Selection of the action to be performed by the application on the infected objects.	`DisinfectDeleteIfNotPossible` (disinfect; delete if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. `DisinfectInformIfNotPossible` (disinfect; inform if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. `Inform` means the application does not try to disinfect or delete the infected object. Information about the infected object is logged. Default value: `DisinfectDeleteIfNotPossible`.
`UseExcludeMasks`	Enables exclusion of the objects specified by the `ExcludeMasks.item_#` setting from the scan.	`Yes` — Exclude objects specified by the `ExcludeMasks.item_#` setting from scans. `No` (default value) — Do not exclude objects specified by the `ExcludeMasks.item_#` setting from the scan.

`ExcludeMasks.item_#`	Excludes objects from being scanned by name or mask. You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format.	The default value is not defined.
		Example:
		`UseExcludeMasks=Yes`
		`ExcludeMasks.item_0000=eicar1.*`
		`ExcludeMasks.item_0001=eicar2.*`

`UseExcludeThreats`	Enables exclusion of objects containing the threats specified by the `ExcludeThreats.item_#` setting from scans.	`Yes` — Exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan. `No` (default value): do not exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan.

`ExcludeThreats.item_#`	Excludes objects from scans by the name of the threats detected in them. Before specifying a value for this setting, make sure that the `UseExcludeThreats` setting is enabled.
	In order to exclude an object from scans, specify the full name of the threat detected in this object – the string containing the application's decision that the object is infected.
	For example, you may be using a utility to obtain information about your network. To keep the application from blocking it, add the full name of the threat contained in it to the list of threats excluded from scans.
	You can find the full name of the threat detected in an object in the application log or on the website https://threats.kaspersky.com.
		The setting value is case-sensitive.
		The default value is not defined.
		Example:
		`UseExcludeThreats=Yes`
		`ExcludeThreats.item_0000=EICAR-Test-*`
		`ExcludeThreats.item_0001=?rojan.Linux`

`UseGlobalExclusions`	Enables global exclusions for scanning.	`Yes` (default value) — Use the global exclusions. `No` — Do not use global exclusions.
`UseOASExclusions`	Enables File Threat Protection exclusions for scanning.	`Yes` (default value) — Use File Threat Protection exclusions. `No` — do not use File Threat Protection exclusions.

Setting	Description	Values
`ReportCleanObjects`	Enables logging of information about scanned objects that the application reports as not being infected. You can enable this setting, for example, to make sure that a particular object was scanned by the application.	`Yes` — Log information about non-infected objects. `No` (default value) — Do not log information about non-infected objects.
`ReportPackedObjects`	Enables logging of information about scanned objects that are part of compound objects. You can enable this setting, for example, to make sure that an object within an archive has been scanned by the application.	`Yes` — Log information about scanned objects within archives. `No` (default value) — Do not log information about scanned objects within archives.
`ReportUnprocessedObjects`	Enables logging of information about objects that have not been processed for some reason.	`Yes` — Log information about unprocessed objects. `No` (default value) — Do not log information about unprocessed objects.
`UseAnalyzer`	Enables heuristic analysis. Heuristic analysis helps the application to detect threats even before they become known to virus analysts.	`Yes` (default value) — Enable Heuristic Analyzer. `No` — Disable Heuristic Analyzer.
`HeuristicLevel`	Specifies the heuristic analysis level. You can specify the heuristic analysis level. The heuristic analysis level sets the balance between the thoroughness of searches for threats, the load on the operating system's resources, and the scan duration. The higher the heuristic analysis level, the more resources and time are required for scanning.	`Light` — The least thorough scan with minimum load on the system. `Medium` — A medium heuristic analysis level with a balanced load on the system. `Deep` — The most thorough scan with maximum load on the system. `Recommended` (default value) — The recommended value.
`UseIChecker`	Enables usage of the iChecker technology.	`Yes` (default value) — Enable use of the iChecker technology. `No` — Disable use of the iChecker technology.
`DeviceNameMasks.item_#`	List of device names. The application will scan boot sectors of these devices. The setting value cannot be empty. At least one device name mask must be specified to run this task.	`AllObjects` – scan boot sectors of all devices. `< device name mask >` – scan boot sectors of the devices whose names match the specified mask. Default value: `/**` – any set of characters in the device name, including the `/` character.
The **[ScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan scope, which contains additional information about the scan scope. The maximum length of the string specified using this setting is 4096 characters.	Default value: `All objects`. Example: `AreaDesc=" Scanning of email databases "`
`UseScanArea`	Enables scans of the specified scope. To run the task, enable scans of at least one scope.	`Yes` (default value) — Scan the specified scope. `No` — Do not scan the specified scope.
`AreaMask.item_#`	Scan scope limitation. Within the scan scope, the application scans only the files that are specified using the masks in the shell format. If this setting is not specified, the application scans all the objects in the scan scope. You can specify several values for this setting.	The default value is `*` (scan all objects). Example: `AreaMask.item_< item number >=*doc`

Setting	Description	Values
`Path`	Path to the directory with objects to be scanned.	`< path to local directory >` — scan objects in the specified directory. `Shared:NFS` — Scan the device file system resources that are accessible via the NFS protocol. `Shared:SMB` – Scan the device file system resources that are accessible via the Samba protocol. `Mounted:NFS` – Scan the remote directories mounted on a device using the NFS protocol. `Mounted:SMB` – Scan the remote directories mounted on a device using the Samba protocol. `AllRemoteMounted` – Scan all remote directories mounted on the device using the Samba and NFS protocols. `AllShared` – Scan all the device file system resources that are accessible via the Samba and NFS protocols. `<file system type>` – Scan all resources of the specified file system device.
The **[ExcludedFromScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan exclusion scope, which contains additional information about the exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from scans.	`Yes` (default value) — Exclude the specified scope. `No` — Do not exclude the specified scope.
`AreaMask.item_#`	Limitation of scan exclusion scope. In the exclusion scope, the application excludes only the files that are specified using masks in the shell format. If this setting is not specified, the application excludes all the objects in the exclusion scope. You can specify several values for this setting.	Default value: `*` (exclude all objects)

Setting	Description	Values
`Path`	Path to the directory with objects to be excluded.	`<path to local directory>` – Exclude objects in the specified directory (including subdirectories) from scans. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. In order to optimize the operation of scan tasks, it is recommended to add the path with snapshots mounted by the system in the read-only mode to the exclusions for the systems with the btrfs file system and enabled active snapshots. For example, for the systems based on SUSE/OpenSUSE, you can add the following exclusion `/.snapshots/*/snapshot/`. The `/` path is specified by default. The application excludes all directories of the local file system from scan. `Mounted:NFS` – Exclude the remote directories mounted on a device using the NFS protocol from scan. `Mounted:SMB` – Exclude the remote directories mounted on a device using the Samba protocol from scan. `AllRemoteMounted` – Exclude all remote directories mounted on the device using the Samba and NFS protocols from scan. `<file system type>` – Exclude from scanning all resources of the specified file system device. Remote directories are excluded from scanning by the application only if they were mounted before the task was started. Remote directories mounted after the task is started are not excluded from scanning.

Critical Areas Scan

When performing a critical areas scan, Kaspersky Embedded Systems Security can scan boot sectors, startup objects, process memory, and kernel memory.

Upon detecting malware, the application can remove the infected file and terminate the malware process started from this file.

> If the root user does not have access to a remote directory mounted on the device over the NFSv4 protocol, the application blocks access to such a file for all users.

You can start a critical areas scan and configure the settings of the scan:

- Select the operating system objects to be scanned. Scanning of boot sectors, process memory and kernel memory, startup objects and archives is enabled by default. By default, files are not scanned during the critical areas scan.
- Limit the size of an object to be scanned and the duration of the object scan.
- Select the actions to be performed by the application on the infected objects.
- Configure exclusions of objects from scans:
  - by name or mask
  - by the name of the threats detected in the objects

  > When running on operating systems that use the special-purpose parsecfs file system (Astra Linux Special Edition) we recommend adding mounting paths of this file system to exclusions from scanning.
- Enable or disable global exclusions and File Threat Protection exclusions when scanning.
- Enable the logging of information about scanned non-infected objects, about scanning objects in archives, and about unprocessed objects.
- Configure the use of the heuristic analyzer and iChecker technology during a scan.
- Limit the set of devices whose boot sectors need to be scanned.
- Configure scan scopes and scan exclusion scopes.

Critical Areas Scan in the Web Console

In the Web Console, you can perform a critical areas scan of the operating system of a protected device using the *Critical Areas Scan* task.

You can create and run critical areas scan user tasks. You can configure the scan settings by editing the settings of the tasks.

Setting	Description
Scan archives	This check box enables or disables scan of archives. If the check box is selected, the application scans the archives. To scan an archive, the application has to unpack it first, which may slow down scanning. You can reduce the duration of archive scans by configuring the **Skip file that is scanned for longer than (sec)** and **Skip file larger than (MB)** settings in the **General scan settings** section. If the check box is cleared, the application does not scan the archives. This check box is selected by default.
Scan SFX archives	This check box enables or disables *self-extracting archive* scans. Self-extracting archives are the archives that contain an executable extraction module. If the check box is selected, the application scans self-extracting archives. If the check box is cleared, the application does not scan self-extracting archives. This check box is available if the **Scan archives** check box is unchecked. This check box is selected by default.
Scan mail databases	This check box enables or disables scans of mail databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail applications. If the check box is selected, the application scans mail database files. If the check box is cleared, the application does not scan mail database files. This check box is cleared by default.
Scan mail format files	This check box enables or disables scan of files of plain-text email messages. If this check box is selected, the application scans plain-text messages. If this check box is cleared, the application does not scan plain-text messages. This check box is cleared by default.
Skip file that is scanned for longer than (sec)	In this field, you can specify the maximum time to scan a file, in seconds. After the specified time, the application stops scanning the file. Possible values: 0–9999. If the value is set to 0, the scan time is unlimited. Default value: 0.
Skip file larger than (MB)	In this field, you can specify the maximum size of a file to scan, in megabytes. Possible values: 0–999999. If the value is set to 0, the application scans files of any size. Default value: 0.
Log clean objects	This check box enables or disables the logging of *ObjectProcessed* type events. If this check box is selected, the application logs events of the *ObjectProcessed* type for all scanned objects. If this check box is cleared, the application does not log events of the *ObjectProcessed* type for any scanned object. This check box is cleared by default.
Notify about unprocessed files	This check box enables or disables the logging *ObjectNotProcessed* type events if a file cannot be processed during a scan. If this check box is selected, the application logs the events of the *ObjectNotProcessed* type. If this check box is cleared, the application does not log the events of the *ObjectNotProcessed* type. This check box is cleared by default.
Log packed objects	This check box enables or disables the logging of *PackedObjectDetected* type events for all packed objects that are detected. If this check box is selected, the application logs the events of the *PackedObjectDetected* type. If this check box is cleared, the application does not log the events of the *PackedObjectDetected* type. This check box is cleared by default.
Use iChecker technology	This check box enables or disables scan of only new and modified since the last scan files. If the check box is selected, the application scans only new files or the files modified since the last scan. If the check box is cleared, the application scans the files regardless of the creation or modification date. This check box is selected by default.
Use heuristic analysis	This check box enables or disables heuristic analysis during file scans. This check box is selected by default.

Setting	Description
Heuristic analysis level	If the **Use heuristic analysis** check box is selected, you can select the heuristic analysis level in the drop-down list: • **Light** is the least detailed scan with minimal system load. • **Medium** is a medium scan with balanced system load. • **Deep** is the most detailed scan with maximum system load. • **Recommended** (default value) is the optimal level recommended by Kaspersky experts. It ensures an optimal combination of protection quality and impact on the performance of the protected devices.
Action on threat detection	You can select the action that the application performs on a detected infected object: • **Disinfect; delete if disinfection is not possible** (default). If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. • **Disinfect; inform if disinfection fails**. If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. • **Inform**. If you select this action, the application creates a record about the infected object in the log.
Scan scopes	The table that contains the scopes scanned by the task. By default, the table contains one scan scope that includes all directories of the local file system. You can add, modify, delete, move up, or move down scan scopes in the table. You can also import the list of scan scopes from a file by clicking **Import** or export the list of scan scopes to a file in JSON format by clicking **Export**. When importing, you are prompted to replace the list of scan scopes or add scan scopes to the existing list.

Add scan scope window

In this window, you can add and configure scan scopes.

Scan scope settings

Setting	Description
Scope name	Field for entering the scan scope name. This name is displayed in the **Scan scopes** table in the **Scan settings** section. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this scan scope. If this check box is cleared, the application does not process this scan scope. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	You can select the type of file system in the drop-down list: • **Local** (default value) – local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
Access protocol	You can select the remote access protocol in the drop-down list: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below. This drop-down list is available if the **Shared** or **Mounted** type is selected in the drop-down list of file systems.

Setting	Description
Path	This is the entry field for specifying the path to the directory that you want to include in the scan scope. You can use **masks** to specify the path. You can use the * (asterisk) character to create a file or directory name mask. You can indicate a single * character to represent any set of characters (including an empty set) preceding the / character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive * characters to represent any set of characters (including an empty set and the / character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The ** mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single ? character to represent any one character in the file or directory name. The **/** path is specified by default – the application scans all directories of the local file system. This field is available if the **Local** type is selected in the drop-down list of file systems. If the **Local** type is selected in the drop-down list of file systems, and the path is not specified, the application scans all directories of the local file system.
Name of shared resource	The field for entering the name of the file system shared resource, where the directories that you want to add to the scan scope are located. The field is available if the **Mounted** type is selected in the File system drop-down list and the **Custom** item is selected in the **Access protocol** drop-down list.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks.

Scan scopes section

Scan scope settings for the Critical area scan task

Setting	Description
Scan files	This check box enables or disables file scans. If the check box is selected, the application scans the files. If the check box is cleared, the application does not scan the files. This check box is cleared by default.
Scan boot sectors	This check box enables or disables boot sector scans. If the check box is selected, the application scans the boot sectors. If the check box is cleared, the application does not scan the boot sectors. This check box is selected by default.
Scan kernel memory and running processes	This check box enables or disables client device memory scan. If the check box is selected, the application scans kernel memory and running processes. If the check box is cleared, the application does not scan kernel memory and running processes. This check box is selected by default.
Scan startup objects	This check box enables or disables startup object scans. If the check box is selected, the application scans startup objects. If the check box is cleared, the application does not scan startup objects. This check box is selected by default.
Devices to scan	Clicking the **Configure device masks** link opens the **Scan scopes** window, where you can specify the devices whose boot sectors will be scanned.

Scan scopes window

The table contains name masks of the devices, whose boot sectors the application must scan. By default, the table contains the **/**** device name mask (all devices).

You can add, edit, and delete items in the table.

Exclusion scopes section

In the **Exclusion scopes** section for the Critical Areas Scan task, you can configure exclusion scopes, exclusions by mask and by threat name, as well as the use of global exclusions and File Threat Protection exclusions when the task is running.

Settings of scan exclusions

Setting	Description
Configure exclusion scopes	Clicking the **Configure exclusion scopes** link opens the **Exclusion scopes** window. In this window, you can define the list of scan exclusions.
Configure exclusions by mask	Clicking the **Configure exclusions by mask** link opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.
Configure exclusions by threat name	Clicking the **Configure exclusions by threat name** link opens the **Exclusions by threat name** window. In this window, you can configure the exclusion of objects from scans based on threat name.
Use global exclusions	The check box enables or disables the exclusion of the mount points specified in global exceptions while the application is running. If this check box is selected, the application excludes configured mount points from scans. This check box is selected by default.
Use File Threat Protection exclusions	This check box enables or disables the use of configured File Threat Protection exclusions when the application is running. If the check box is selected, the application does not scan the objects specified in the exclusions for the File Threat Protection component. This check box is selected by default.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

Add exclusion scope window

In this window, you can add and configure exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the Exclusion scopes window. The entry field must not be blank.
Use this scope	This check box enables or disables the exclusion of the scope when the application is running. If the check box is selected, the application excludes this scope from scan or protection during its operation. If the check box is cleared, the application includes this scope in scan or protection during its operation. You can later exclude this scope from scan or protection by selecting the check box. This check box is selected by default.
File system, access protocol, and path	In this drop-down list, you can select the type of file system where the directories that you want to add to scan exclusions are located: • **Local**, for local directories. • **Mounted**, for remote directories mounted on the device. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
Access protocol	You can select the remote access protocol in the drop-down list: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below. This drop-down list is available if the **Mounted** type is selected in the drop-down list of file systems.
Path	Entry field for the path to the directory that you want to add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan. This field is available if the **Local** type is selected in the drop-down list of file systems.
Name of shared resource	The field for entering the name of the file system shared resource, where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the File system drop-down list and the **Custom** item is selected in the **Access protocol** drop-down list.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the **Path** field. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** or **Edit** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by threat name window

You can configure the exclusion of objects from scans based on threat name. The application will not block the specified threats. By default, the list of threat names is empty.

You can add, edit, and delete threat names.

Clicking the **Add** or **Edit** button opens the **Threat name** window. In this window, you can define the name of the threat to be excluded from a scan.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Critical Areas Scan in the Administration Console

In the Administration Console, you can perform a critical areas scan of the operating system of a protected device using the *Critical Areas Scan* task.

You can create and run critical areas scan user tasks. You can configure the scan settings by editing the settings of the tasks.

In the **Settings** section of the properties of the Critical Areas Scan task, you can configure the settings listed in the table below.

Critical Areas Scan task settings

Setting	Description
Scan	This group of settings contains buttons that open windows where you can configure the scan scopes, scan scope settings, and scan settings.
Actions on threat detection	This group of settings contains the **Configure** button. Clicking this button opens the **Actions on threat detection** window, where you can configure the actions that the application performs on detected infected objects.

In the **Exclusions** section, you can configure exclusion scopes as well as exclusions by mask and by the threat name in the properties of the Critical Areas Scan task.

Scan scopes window

The table contains the scan scopes. The application will scan files and directories located in the paths specified in the table. By default, the table contains one scan scope that includes all directories of the local file system.

Scan scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Scope name	Scan scope name.
Path	Path to the directory that the application scans.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure scan scopes.

Scan scope settings

Setting	Description
Scan scope name	Field for entering the scan scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this scan scope. If this check box is cleared, the application does not process this scan scope. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	The settings block lets you set the scan scope. You can select the file system type in the drop-down list of file systems: • **Local** (default value) – local directories. If this item is selected, you need to indicate the path to the local directory. • **Mounted** – Mounted remote or local directories. If this item is selected, you need to indicate the protocol or name of the file system. • **Shared** — The protected server's file system resources accessible via the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols. • **All shared** — All of the protected server's file system resources accessible via the Samba and NFS protocols.
	If **Shared** or **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want to add to the scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default – the application scans all directories of the local file system. If the **Local** type is selected in the drop-down list of file systems, and the path is not specified, the application scans all directories of the local file system.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the scan scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Scan scope settings window

In this window, you can configure the scan settings for the Critical Areas Scan task. The application allows you to scan files, boot sectors, startup objects, process memory, and kernel memory.

Scan scope settings

Setting	Description
Scan files	This check box enables or disables file scans. If this check box is selected, Kaspersky Embedded Systems Security will scan files. If this check box is unchecked, Kaspersky Embedded Systems Security will not scan files. This check box is cleared by default.
Scan boot sectors	This check box enables or disables boot sector scans. If this check box is selected, Kaspersky Embedded Systems Security will scan boot sectors. If this check box is unchecked, Kaspersky Embedded Systems Security will not scan boot sectors. This check box is selected by default.

Setting	Description
Scan kernel memory and running processes	This check box enables or disables device memory scan. If the check box is selected, Kaspersky Embedded Systems Security scans kernel memory and running processes. If the check box is cleared, Kaspersky Embedded Systems Security does not scan kernel memory and running processes. This check box is selected by default.
Scan startup objects	This check box enables or disables startup object scans. If this check box is selected, Kaspersky Embedded Systems Security will scan startup objects. If this check box is unchecked, Kaspersky Embedded Systems Security will not scan startup objects. This check box is selected by default.
Devices to scan	This group of settings contains the **Configure** button. Clicking this button opens the **Scan scopes** window, where you can specify the devices whose boot sectors must be scanned.
Use global exclusions	The check box enables or disables the exclusion of the mount points specified in global exceptions while the application is running. If this check box is selected, the application excludes configured mount points from scans. This check box is selected by default.
Use File Threat Protection exclusions	This check box enables or disables the use of configured File Threat Protection exclusions when the application is running. If the check box is selected, the application does not scan the objects specified in the exclusions for the File Threat Protection component. This check box is selected by default.

Scan scopes window

The table contains name masks of the devices, whose boot sectors the application must scan. By default, the table contains the **/**** device name mask (all devices).

You can add, edit, and delete items in the table.

Scan settings window

In this window, you can configure the file scan settings for the task.

Scan settings

Setting	Description
Scan archives	This check box enables or disables scan of archives. If the check box is selected, the application scans the archives. To scan an archive, the application has to unpack it first, which may slow down scanning. You can reduce the duration of archive scans by configuring the **Skip file that is scanned for longer than (sec)** and **Skip file larger than (MB)** settings in the **General scan settings** section. If the check box is cleared, the application does not scan the archives. This check box is selected by default.
Scan SFX archives	This check box enables or disables *self-extracting archive* scans. Self-extracting archives are the archives that contain an executable extraction module. If the check box is selected, the application scans self-extracting archives. If the check box is cleared, the application does not scan self-extracting archives. This check box is available if the **Scan archives** check box is unchecked. This check box is selected by default.

Setting	Description
Scan mail databases	This check box enables or disables scans of mail databases of Microsoft Outlook, Outlook Express, The Bat!, and other mail applications. If the check box is selected, the application scans mail database files. If the check box is cleared, the application does not scan mail database files. This check box is cleared by default.
Scan mail format files	This check box enables or disables scan of files of plain-text email messages. If this check box is selected, the application scans plain-text messages. If this check box is cleared, the application does not scan plain-text messages. This check box is cleared by default.
Skip file that is scanned for longer than (sec)	In this field, you can specify the maximum time to scan a file, in seconds. After the specified time, the application stops scanning the file. Possible values: 0–9999. If the value is set to 0, the scan time is unlimited. Default value: 0.
Skip file larger than (MB)	In this field, you can specify the maximum size of a file to scan, in megabytes. Possible values: 0–999999. If the value is set to 0, the application scans files of any size. Default value: 0.
Log clean objects	This check box enables or disables the logging of *ObjectProcessed* type events. If this check box is selected, the application logs events of the *ObjectProcessed* type for all scanned objects. If this check box is cleared, the application does not log events of the *ObjectProcessed* type for any scanned object. This check box is cleared by default.
Log unprocessed objects	This check box enables or disables the logging *ObjectNotProcessed* type events if a file cannot be processed during a scan. If this check box is selected, the application logs the events of the *ObjectNotProcessed* type. If this check box is cleared, the application does not log the events of the *ObjectNotProcessed* type. This check box is cleared by default.
Log packed objects	This check box enables or disables the logging of *PackedObjectDetected* type events for all packed objects that are detected. If this check box is selected, the application logs the events of the *PackedObjectDetected* type. If this check box is cleared, the application does not log the events of the *PackedObjectDetected* type. This check box is cleared by default.
Use iChecker technology	This check box enables or disables scan of only new and modified since the last scan files. If the check box is selected, the application scans only new files or the files modified since the last scan. If the check box is cleared, the application scans the files regardless of the creation or modification date. This check box is selected by default.
Use heuristic analysis	This check box enables or disables heuristic analysis during file scans. This check box is selected by default.
Heuristic analysis level	If the **Use heuristic analysis** check box is selected, you can select the heuristic analysis level in the drop-down list: • **Light** is the least detailed scan with minimal system load. • **Medium** is a medium scan with balanced system load. • **Deep** is the most detailed scan with maximum system load. • **Recommended** (default value) is the optimal level recommended by Kaspersky experts. It ensures an optimal combination of protection quality and impact on the performance of the protected devices.

Actions on threat detection window

In this window, you can configure actions to be performed by Kaspersky Embedded Systems Security on detected infected objects:

Actions on threat detection

Setting	Description
Actions on threat detection	You can select the action that the application performs on a detected infected object: • **Disinfect; delete if disinfection is not possible** (default). If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. • **Disinfect; inform if disinfection fails**. If you select this action, the application attempts to disinfect the infected object by saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. • **Inform**. If you select this action, the application creates a record about the infected object in the log.

Exclusions section

Scan exclusion is a set of conditions. When these conditions are met, Kaspersky Embedded Systems Security does not scan the objects for viruses and other malware. You can also exclude objects from scans by masks and threat names.

Settings of scan exclusions

Group of settings	Description
Exclusion scopes	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from scans.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.
Exclusions by threat name	This group of settings contains the **Configure** button, which opens the **Exclusions by threat name** window. In this window, you can configure the exclusion of objects from scans based on threat name.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

<New exclusion scope> window

In this window, you can add and configure scan exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables exclusion of the scope from scan when the application is running. If this check box is selected, the application excludes this area during scans. If this check box is cleared, the application includes this area in the scan scope. You can later exclude this scope by selecting the check box. This check box is selected by default.
File system, access protocol, and path	The settings block lets you set the exclusion scope. In the drop-down list of file systems, you can select the type of file system of the directories to be excluded from scans: • **Local**, for local directories. • **Mounted** – mounted directories. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
	If **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the path field. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Exclusions by threat name window

You can configure the exclusion of objects from scans based on threat name. The application will not block the specified threats. By default, the list of threat names is empty.

You can add, edit, and delete threat names.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Critical Areas Scan on the command line

In the command line, you can perform a critical areas scan of the operating system of a protected device using the Critical Areas Scan predefined task (*Critical_Areas_Scan*).

You can manually start, stop, pause, or resume this task and configure the task run schedule. You can configure scan settings by editing the settings of this task.

Critical Areas Scan task settings

Setting	Description	Values
`ScanFiles`	Enables file scan.	`Yes` — Scan files. `No` (default value) — Do not scan files.
`ScanBootSectors`	Enables boot sector scans.	`Yes` (default value) — Scan boot sectors. `No` — Do not scan boot sectors.

Setting	Description	Values
`ScanComputerMemory`	Enables process memory and kernel memory scans.	`Yes` (default value) — Scan process memory and kernel memory. `No` — Do not scan process memory and kernel memory.
`ScanStartupObjects`	Enables startup object scans.	`Yes` (default value) — Scan startup objects. `No` — Do not scan startup objects.
`ScanArchived`	Enables scanning of archives (including SFX self-extracting archives). The application scans the following archives: .zip; .7z*; .7-z; .rar; .iso; .cab; .jar; .bz; .bz2; .tbz; .tbz2; .gz; .tgz; .arj. The list of supported archive formats depends on the application databases being used.	`Yes` (default value) — Scan archives. If `ActionOnThreat` is set to `DisinfectDeleteIfNotPossible`, and disinfection is impossible, then, depending on the archive type, the application deletes either the infected object or the entire archive that contains the threat. `No` — Do not scan archives.
`ScanSfxArchived`	Enables scanning of self-extracting archives only (archives that contain an executable extraction module).	`Yes` (default value) — Scan self-extracting archives. `No` — Do not scan self-extracting archives.
`ScanMailBases`	Enables scanning email databases of Microsoft Outlook, Outlook Express, The Bat, and other mail clients.	`Yes` — Scan files of email databases. `No` (default value) — Do not scan files of email databases.
`ScanPlainMail`	Enables scanning of plain text email messages.	`Yes` — Scan plain text email messages. `No` (default value) — Do not scan plain text email messages.
`SizeLimit`	Maximum size of an object to be scanned (in megabytes). If the object to be scanned is larger than the specified value, the application skips this object.	0 to 999999 0 — The application scans objects of any size. Default value: 0.
`TimeLimit`	Maximum object scan duration (in seconds). The application stops scanning the object if it takes longer than the time specified by this setting.	0 to 9999 0 — The object scan time is unlimited. Default value: 0.
`ActionOnThreat`	Selection of the action to be performed by the application on the infected objects.	`DisinfectDeleteIfNotPossible` (disinfect; delete if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), then the application deletes the infected object. `DisinfectInformIfNotPossible` (disinfect; inform if not possible) means the application tries to disinfect the infected object after saving a copy of the object in Backup. If disinfection is not possible (for example, if the type of object or the type of threat in the object cannot be disinfected), the application creates a record in the log. `Inform` means the application does not try to disinfect or delete the infected object. Information about the infected object is logged. Default value: `DisinfectDeleteIfNotPossible`.
`UseExcludeMasks`	Enables exclusion of the objects specified by the `ExcludeMasks.item_#` setting from the scan.	`Yes` — Exclude objects specified by the `ExcludeMasks.item_#` setting from scans. `No` (default value) — Do not exclude objects specified by the `ExcludeMasks.item_#` setting from the scan.

`ExcludeMasks.item_#`	Excludes objects from being scanned by name or mask. You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format.
	Before specifying a value for this setting, make sure that the `UseExcludeMasks` setting is enabled.
		The default value is not defined.
		Example:
		`UseExcludeMasks=Yes`
		`ExcludeMasks.item_0000=eicar1.*`
		`ExcludeMasks.item_0001=eicar2.*`

Setting	Description	Values
`UseExcludeThreats`	Enables exclusion of objects containing the threats specified by the `ExcludeThreats.item_#` setting from scans.	`Yes` — Exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan. `No` (default value): do not exclude objects containing the threats specified by the `ExcludeThreats.item_#` setting from the scan.

`ExcludeThreats.item_#`	Excludes objects from scans by the name of the threats detected in them. Before specifying a value for this setting, make sure that the `UseExcludeThreats` setting is enabled.
	In order to exclude an object from scans, specify the full name of the threat detected in this object – the string containing the application's decision that the object is infected.
	For example, you may be using a utility to obtain information about your network. To keep the application from blocking it, add the full name of the threat contained in it to the list of threats excluded from scans.
	You can find the full name of the threat detected in an object in the application log or on the website https://threats.kaspersky.com.
		The setting value is case-sensitive.
		The default value is not defined.
		Example:
		`UseExcludeThreats=Yes`
		`ExcludeThreats.item_0000=EICAR-Test-*`
		`ExcludeThreats.item_0001=?rojan.Linux`

`UseGlobalExclusions`	Enables global exclusions for scanning.	`Yes` (default value) — Use the global exclusions. `No` — Do not use global exclusions.
`UseOASExclusions`	Enables File Threat Protection exclusions for scanning.	`Yes` (default value) — Use File Threat Protection exclusions. `No` — do not use File Threat Protection exclusions.
`ReportCleanObjects`	Enables logging of information about scanned objects that the application reports as not being infected. You can enable this setting, for example, to make sure that a particular object was scanned by the application.	`Yes` — Log information about non-infected objects. `No` (default value) — Do not log information about non-infected objects.
`ReportPackedObjects`	Enables logging of information about scanned objects that are part of compound objects. You can enable this setting, for example, to make sure that an object within an archive has been scanned by the application.	`Yes` — Log information about scanned objects within archives. `No` (default value) — Do not log information about scanned objects within archives.
`ReportUnprocessedObjects`	Enables logging of information about objects that have not been processed for some reason.	`Yes` — Log information about unprocessed objects. `No` (default value) — Do not log information about unprocessed objects.
`UseAnalyzer`	Enables heuristic analysis. Heuristic analysis helps the application to detect threats even before they become known to virus analysts.	`Yes` (default value) — Enable Heuristic Analyzer. `No` — Disable Heuristic Analyzer.
`HeuristicLevel`	Specifies the heuristic analysis level. You can specify the heuristic analysis level. The heuristic analysis level sets the balance between the thoroughness of searches for threats, the load on the operating system's resources, and the scan duration. The higher the heuristic analysis level, the more resources and time are required for scanning.	`Light` — The least thorough scan with minimum load on the system. `Medium` — A medium heuristic analysis level with a balanced load on the system. `Deep` — The most thorough scan with maximum load on the system. `Recommended` (default value) — The recommended value.
`UseIChecker`	Enables usage of the iChecker technology.	`Yes` (default value) — Enable use of the iChecker technology. `No` — Disable use of the iChecker technology.
`DeviceNameMasks.item_#`	List of device names. The application will scan boot sectors of these devices. The setting value cannot be empty. At least one device name mask must be specified to run this task.	`AllObjects` – scan boot sectors of all devices. `< device name mask >` – Scan boot sectors of the devices whose names match the specified mask. Default value: `/**` – any set of characters in the device name, including the `/` character.

Setting	Description	Values
The **[ScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan scope, which contains additional information about the scan scope. The maximum length of the string specified using this setting is 4096 characters.	Default value: `All objects`. Example: `AreaDesc=" Mail bases scan "`
`UseScanArea`	Enables scans of the specified scope. To run the task, enable scans of at least one scope.	`Yes` (default value) — Scan the specified scope. `No` — Do not scan the specified scope.
`AreaMask.item_#`	Scan scope limitation. Within the scan scope, the application scans only the files that are specified using the masks in the shell format. If this setting is not specified, the application scans all the objects in the scan scope. You can specify several values for this setting.	The default value is `*` (scan all objects). Example: `AreaMask.item_< item number >=*doc`
`Path`	Path to the directory with objects to be scanned.	`< path to local directory >` — scan objects in the specified directory. `Shared:NFS` — Scan the device file system resources that are accessible via the NFS protocol. `Shared:SMB` – Scan the device file system resources that are accessible via the Samba protocol. `Mounted:NFS` – Scan the remote directories mounted on a device using the NFS protocol. `Mounted:SMB` – Scan the remote directories mounted on a device using the Samba protocol. `AllRemoteMounted` – Scan all remote directories mounted on the device using the Samba and NFS protocols. `AllShared` – Scan all the device file system resources that are accessible via the Samba and NFS protocols. `<file system type>` – Scan all resources of the specified file system device.
The **[ExcludedFromScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the scan exclusion scope, which contains additional information about the exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from scans.	`Yes` (default value) — Exclude the specified scope. `No` — Do not exclude the specified scope.
`AreaMask.item_#`	Limitation of scan exclusion scope. In the exclusion scope, the application excludes only the files that are specified using masks in the shell format. If this setting is not specified, the application excludes all the objects in the exclusion scope. You can specify several values for this setting.	Default value: `*` (exclude all objects)

Setting	Description	Values
`Path`	Path to the directory with objects to be excluded.	`<path to local directory>` – Exclude objects in the specified directory (including subdirectories) from scans. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. In order to optimize the operation of scan tasks, it is recommended to add the path with snapshots mounted by the system in the read-only mode to the exclusions for the systems with the btrfs file system and enabled active snapshots. For example, for the systems based on SUSE/OpenSUSE, you can add the following exclusion `/.snapshots/*/snapshot/`. The `/` path is specified by default. The application excludes all directories of the local file system from scan. `Mounted:NFS` – Exclude the remote directories mounted on a device using the NFS protocol from scan. `Mounted:SMB` – Exclude the remote directories mounted on a device using the Samba protocol from scan. `AllRemoteMounted` – Exclude all remote directories mounted on the device using the Samba and NFS protocols from scan. `<file system type>` – Exclude from scanning all resources of the specified file system device. Remote directories are excluded from scanning by the application only if they were mounted before the task was started. Remote directories mounted after the task is started are not excluded from scanning.

Removable drives scan

Kaspersky Embedded Systems Security can scan the following removable drives when they are connected to the protected device: CDs, DVDs, Blu-ray discs, flash drives (including USB modems), external hard drives, and floppy disks.

If the removable drives scan is enabled, Kaspersky Embedded Systems Security monitors the connection of removable drives to the protected device and, if a connected removable drive is detected, it scans the drive and its boot sectors for viruses and other malware.

By default, the application does not monitor for the connection of removable drives or scan these.

Configuring Removable Drives Scan in the Web Console

In the Web Console, you can manage the settings of the removable drives scan in the policy properties (**Application settings** → **Host Threat Protection** → **Removable Drives Scan**).

Removable Drives Scan component settings

Setting	Description
Scan type	**Quick Scan**. If this option is selected, after a removable drive is connected, Kaspersky Embedded Systems Security scans only files of certain formats that are most susceptible to infection, and does not unpack compound objects. A quick scan is performed using the default settings for the *Critical Areas Scan* task. **Detailed Scan**. If this option is selected, after a removable drive is connected, Kaspersky Embedded Systems Security scans all files located on the removable drive. A detailed scan is performed using the default settings for the *Malware Scan* task.
Block access to the removable drive while scanning	If this check box is selected, Kaspersky Embedded Systems Security blocks access to files on the connected drive for the duration of the scan.

Configuring Removable Drives Scan in the Administration Console

In the Administration Console, you can manage Removable Drives Scan settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Local Tasks** → **Removable Drives Scan**).

Removable Drives Scan component settings

Setting	Description
Enable removable drives scan when connected to the device	This check box enables or disables the scan of removable drives when they are connected to the user device. This check box is cleared by default.
Action when a removable drive connects	In the drop-down list, you can select an action to be performed by the application upon connection of removable drives to the user device: • **Do not scan** removable drives when connected (default value). • **Quick scan** – only scan files of **certain types** on removable drives (except CD/DVD drives and Blu-ray discs) and do not unpack compound objects. A quick scan is performed using the default settings for the *Critical Areas Scan* task. The following file formats are scanned on removable media: com, exe, sys, prg, bin, bat, cmd, dpl, dll, scr, cpl, ocx, tsp, drv, vxd, pif, lnk, reg, ini, cia, vbs, vbe, js, jse, htm, htt, hta, asp, chm, pht, wsh, wsf, the, hip, eml, nws, msg, pig, mbx, doc*, dot*, fpm, rtf, shs, dwg, msi, otm, pdf, swf, jpeg, emf, ico, ov?, xl*, xlsb, pp*, md*, sldx, sldm, thmx. • **Detailed scan** – scan all files on removable drives (except CD/DVD drives and Blu-ray discs). A detailed scan is performed using the default settings for the *Malware Scan* task.
Action when a CD / DVD drive connects	In the drop-down list, you can select an action to be performed by the application upon connection of CD/DVD drives and Blu-ray discs to the user device: • **Do not scan** CD/DVD drives and Blu-ray discs when connected (default value). • **Quick scan**: only scans files of **certain types** on CD/DVD drives and Blu-ray discs. A quick scan is performed using the default settings for the *Critical Areas Scan* task. The following file formats are scanned on removable media: com, exe, sys, prg, bin, bat, cmd, dpl, dll, scr, cpl, ocx, tsp, drv, vxd, pif, lnk, reg, ini, cia, vbs, vbe, js, jse, htm, htt, hta, asp, chm, pht, wsh, wsf, the, hip, eml, nws, msg, pig, mbx, doc*, dot*, fpm, rtf, shs, dwg, msi, otm, pdf, swf, jpeg, emf, ico, ov?, xl*, xlsb, pp*, md*, sldx, sldm, thmx. • **Detailed scan** – scan all files on CD/DVD drives and Blu-ray discs. A detailed scan is performed using the default settings for the *Malware Scan* task.
Block access to the removable drive while scanning	This check box enables or disables blocking of files on the connected drive during execution of the scan. This check box is cleared by default.

Configuring Removable Drives Scan on the command line

On the command line, you can manage removable media scanning using the *Removable Drives Scan* predefined task (*Removable_Drives_Scan*).

Removable Drives Scan is not running by default. You can start and stop this task manually. You can configure scan settings by editing the settings of this task.

If the task is running, the application monitors the connection of removable drives to the device and, when a removable drive is connected, it creates and starts a temporary boot sector scan task (task of the *ODS* type). This task cannot be stopped. After the task execution completes, the application automatically deletes the task.

If you enabled file scanning in the *Removable Drives Scan* task settings, the application also starts one or more temporary custom file scan tasks (tasks of the *ODS* type). If necessary, a user with administrator privileges can stop these tasks.

If you modify the *Removable Drives Scan* task settings, the new values are not applied to temporary tasks that are already running. Stopping the *Removable Drives Scan* task does not stop temporary tasks that are already running.

Removable Drives Scan task settings

Setting	Description	Values
`ScanRemovableDrives`	Enables the scanning of removable media when it is connected to the device. This setting does not apply to CD/DVD drives and Blu-ray discs (see the `ScanOpticalDrives` setting).	`DetailedScan` — Scan all files on removable media (except CD/DVD and Blu-ray discs). A detailed scan is performed with the default settings for the *Scan_File* task (ID: 3). `QuickScan` — Scan only files of **specific types** on removable media (except CD/DVD and Blu-ray discs). The following file formats are scanned on removable media: com, exe, sys, prg, bin, bat, cmd, dpl, dll, scr, cpl, ocx, tsp, drv, vxd, pif, lnk, reg, ini, cia, vbs, vbe, js, jse, htm, htt, hta, asp, chm, pht, wsh, wsf, the, hip, eml, nws, msg, pig, mbx, doc*, dot*, fpm, rtf, shs, dwg, msi, otm, pdf, swf, jpeg, emf, ico, ov?, xl*, xlsb, pp*, md*, sldx, sldm, thmx. A quick scan is performed with the default settings for the *Critical_Areas_Scan* task (ID: 4). `NoScan` (default value) — Do not scan removable media when it is connected to the device.
`ScanOpticalDrives`	Enables the scanning of CD/DVD drives and Blu-ray discs when they are connected to the device.	`DetailedScan` — Scan all files on CD/DVD drives and Blu-ray discs. A detailed scan is performed with the default settings for the *Scan_File* task (ID: 3). `QuickScan` — Scan only files of **specific types** on CD/DVD drives and Blu-ray discs. The following file formats are scanned on removable media: com, exe, sys, prg, bin, bat, cmd, dpl, dll, scr, cpl, ocx, tsp, drv, vxd, pif, lnk, reg, ini, cia, vbs, vbe, js, jse, htm, htt, hta, asp, chm, pht, wsh, wsf, the, hip, eml, nws, msg, pig, mbx, doc*, dot*, fpm, rtf, shs, dwg, msi, otm, pdf, swf, jpeg, emf, ico, ov?, xl*, xlsb, pp*, md*, sldx, sldm, thmx. A quick scan is performed with the default settings for the *Critical_Areas_Scan* task (ID: 4). `NoScan` (default value) — Do not scan CD/DVD drives and Blu-ray discs when they are connected.
`BlockDuringScan`	Enables the blocking of files on a connected disk during scanning. While scanning boot sectors, files are not blocked.	`Yes` — Block files during the scan. `No` (default value) — Do not block files during the scan.

Firewall Management

A device used on local area networks (LANs) and the internet is exposed to viruses, other malware, and a variety of attacks that exploit vulnerabilities in operating systems and software. The operating system firewall protects data stored on the user device by blocking most threats when the device is connected to the internet or a LAN.

The operating system firewall allows you to detect all network connections on the user device and provide a list of their IP addresses. The Firewall Management component allows you to set the status of the network connections by configuring the network packet rules.

You can use network packet rules to specify the desired level of device protection, from fully blocking Internet access for all applications to allowing unlimited access. All outbound connections are allowed by default, unless corresponding blocking rules for the Firewall Management component are specified.

The Firewall Management component is disabled by default.

It is recommended to disable other operating system firewall management tools before enabling the Firewall Management component.

When the Firewall Management component is enabled, Kaspersky Embedded Systems Security automatically deletes all custom rules configured for the firewall with tools provided by the operating system. These rules are not restored after the component is disabled. If required, save the custom firewall rules before enabling the Firewall Management component.

If firewall management is enabled, Kaspersky Embedded Systems Security scans the operating system firewall and blocks any attempt to change the firewall settings, for example, when an application or utility attempts to add or delete a firewall rule. Kaspersky Embedded Systems Security checks the operating system firewall every 60 seconds and, if necessary. restores the set of firewall rules created using the application. The checking period cannot be changed.

Kaspersky Embedded Systems Security still scans the operating system firewall when firewall management is disabled. This allows the application to restore dynamic rules.

You can enable or disable firewall management, and also configure the following settings:

- Configure a list of network packet rules that Kaspersky Embedded Systems Security will apply when an attempt to establish a network connection is detected. You can add or remove network packet rules, and also change the execution priority of a network packet rule.

  In some operating systems based on the Red Hat Enterprise Linux code base, firewall rules created in the Kaspersky Embedded Systems Security application can only be viewed using a management command (`kess-control -F --query`).

- Select default actions to perform on incoming connections and packets if no other network packet rules apply to this connection type.

- Map network addresses to predefined networks. You can add IP addresses or subnets to available networks and delete address from available networks.

- Enables or disables automatic adding of allowing rules for Network Agent ports.

About dynamic rules

Kaspersky Embedded Systems Security allows *dynamic rules* to be added to, or deleted from, the firewall to ensure the application works properly. For example, Network Agent adds dynamic rules that allow connections to Kaspersky Security Center initiated by the application or by Kaspersky Security Center. The rules of the Anti-Cryptor are also dynamic.

Kaspersky Embedded Systems Security does not control dynamic rules and does not block application components' access to network resources. Dynamic rules do not depend on the Firewall Management component state (enabled/disabled) or changes to the settings of the component operation. The execution priority of dynamic rules is higher than the priority of network packet rules. The application restores a set of dynamic rules if any of them are deleted, for example, by using the iptables utility.

You can view the set of dynamic rules (using the `kess-control -F --query` command); however the dynamic rules settings cannot be modified.

About network packet rules

Network packet rules are actions taken by the Kaspersky Embedded Systems Security to allow or deny a detected network connection attempt.

Network packet rules impose restrictions on network packets regardless of the application. Such rules restrict inbound and outbound network traffic through specific ports of the selected data protocol.

All outbound connections are allowed by default (default action setting), unless the corresponding blocking rules for the Firewall Management are specified. The default action is performed with the lowest priority: if no other network packet rule has been triggered or if no network packet rules have been specified, the connection is allowed.

Firewall Management specifies certain network packet rules by default. You can create your own network packet rules and specify an execution priority for each network packet rule.

Network packet rule settings

Setting	Description
Rule name	The field for entering the name of the network packet rule.
Priority	Priority of the network packet rule.
Action	In the drop-down list, you can select an action to be performed by the Firewall Management component when it detects network activity: • **Allow** network activity (default value). • **Block** network activity.
Direction	In this drop-down list, you can specify the direction of the monitored network activity: • **Incoming**. If this option is selected, the Firewall Management component monitors only incoming connections. • **Inbound (packet)**. If this option is selected, the Firewall Management component monitors only incoming packets. • **Incoming / Outgoing**. If this option is selected, the Firewall Management component monitors both incoming and outgoing connections. • **Outbound (packet)**. If this option is selected, the Firewall Management component monitors only outgoing packets. • **Outgoing**. If this option is selected, the Firewall Management component monitors only outgoing connections. • **Incoming / Outgoing (packets)**. If this option is selected, the Firewall Management component monitors both incoming and outgoing packets.

Setting	Description
Log events	This check box lets you specify whether the actions of the network rule are recorded in the report. If the check box is selected, the application writes the actions of the network rule to the report. If the check box is cleared, the application does not write the actions of the network rule to the report. This check box is cleared by default.
Protocol	In the drop-down list, you can select the type of data transfer protocol for which you want to monitor network activity: • **All** • **TCP** • **UDP** • **ICMPv4** • **ICMPv6** • **IGMP** • **GRE** If you select TCP or UDP, you can specify comma-separated ports of the user device and the remote device, the connection between which you want to monitor. If you select ICMP or ICMPv6, you can specify the type and code of the ICMP packet.
Remote ports	If **TCP** or **UDP** is selected in the **Protocol** drop-down list, you can specify a port or range of ports of remote devices, the connection between which you want to monitor. This field is displayed only if **TCP** or **UDP** data transfer protocol is selected in the **Protocol** drop-down list.
Local ports	If **TCP** or **UDP** is selected in the **Protocol** drop-down list, you can specify a port or range of ports of local devices, the connection between which you want to monitor. This field is displayed only if **TCP** or **UDP** data transfer protocol is selected in the **Protocol** drop-down list.
ICMP type	If **ICMPv4** or **ICMPv6** is selected from the **Protocol** drop-down list, you can specify the ICMP type. The Firewall Management component monitors messages of the specified type sent by the host or gateway. If this check box is selected, the field for entering the ICMP type is displayed.
ICMP code	If **ICMPv4** or **ICMPv6** is selected from the **Protocol** drop-down list, you can specify the ICMP code. The Firewall Management component monitors messages of the type specified in the field under the **ICMP type** check box, with the code specified in the field under the **ICMP code** check box, and sent by the host or gateway. If this check box is selected, the field for entering the ICMP code is displayed.
Remote network addresses	The network addresses of the remote devices that can send and receive network packets: • **Any address**. If this option is selected, the network rule controls network packets sent and received by remote devices with any IP address. • **Subnet address**. If this option is selected, the network rule controls network packets sent and received by remote devices with the IP addresses associated with the selected network type: Trusted network, **Public network**, or **Local network**. • **Addresses from the list**. If this option is selected, the network rule controls the sending and receiving of network packets by remote devices with the IP addresses specified in the table below. You can specify multiple addresses at once. In this case, enter each address on a new line to make copying easier.
Local	Network addresses of devices that have Kaspersky Embedded Systems Security installed and can send and/or receive network packets: • **Any address**. If this item is selected, the network rule controls sending and receiving of network packets by the devices with Kaspersky Embedded Systems Security installed and with any IP address. • **Addresses from the list**. If this option is selected, the network rule controls the IP addresses of devices with Kaspersky Embedded Systems Security installed, specified in the table below, that can transmit and receive network packets. You can specify multiple addresses at once. In this case, enter each address on a new line to make copying easier.

About the predefined available networks

An *available network* is a defined group of IP addresses or subnets. An available network lets you use the same rules for several IP addresses or subnets without having to create a separate rule for each IP address or subnet. An available network can be used as the value of the "remote address" parameter when creating a network packet rule. Kaspersky Embedded Systems Security provides three predefined networks:

- **Public**. A network that is not protected by an anti-virus application, firewall, or filters (such as networks of internet cafés).
- **Local**. A network whose users have the right of access to files and printers on the device (such as corporate LANs or home networks).
- **Trusted**. A safe network in which the device is not exposed to attacks or unauthorized data access attempts.

You cannot create a new predefined network or delete an existing one, but you can add or remove IP addresses and subnets from a network.

Firewall Management in the Web Console

In the Web Console, you can manage Firewall settings in the policy properties (**Application settings** → **Communication & Web Protection** → **Firewall**).

Firewall Management settings

Setting	Description
Network packet rules	Network packet rules table. Network packet rules serve to impose restrictions on network packets, regardless of the application. Such rules restrict inbound and outbound network traffic through specific ports of the selected data protocol. This table lists predefined network packet rules that are recommended by Kaspersky experts for optimal protection of network traffic of devices. The Firewall Management component sets an execution priority for each network packet rule and processes network packet rules in the order they are listed in the list of network packet rules, from top to bottom. The Firewall Management component finds the first matching network packet rule a network connection and performs an action that either allows or blocks network activity. Next, the Firewall Management component ignores all lower-priority network packet rules for this network connection. Network packet rules have a higher priority than network rules for applications.

Setting	Description
Available networks	Table of information about network connections detected by the Firewall Management component on the user device. By default, the internet has the **Public network** type. You cannot change the type of the Internet. The following network connection types are available: • **Public** The network is not protected by antivirus applications, firewalls, or filters (such as Wi-Fi in a cafe). The application blocks access to files and printers of a device connected to such a network. Outside users also denied access to information via shared folders and remote access to the desktop of this device. The Firewall Management component filters the network activity of each application in accordance with the network rules of that application. • **Local** A network for users with restricted access to files and printers on this device (for example, a corporate LAN or a home network). • **Trusted** A safe network, in which the device is not exposed to attacks and attempts of unauthorized access to data. For networks with this status, the application allows any network activity within this network.
Always add allowing rules for Network Agent ports	This check box enables or disables the automatic adding of allowing rules for Network Agent ports.

Firewall Management in the Administration Console

In the Administration Console, you can manage Firewall Management settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → Firewall Management**).

Firewall Management settings

Setting	Description
Enable Firewall Management	This check box enables or disables Firewall Management. This check box is cleared by default.
Network packet rules	This group of settings contains the **Configure** button. Clicking this button opens the **Network packet rules** window. In this window, you can configure network packet rules that are applied by the Firewall Management component when it detects the network connection attempt.
Available networks	This group of settings contains the **Configure** button. Clicking this button opens the **Available networks** window. In this window, you can configure the list of networks that the Firewall Management component will monitor.
Incoming connections	In this drop-down list, you can select the action to be performed for incoming network connections: • **Allow** network connections (default value). • **Block** network connections.
Incoming packets	In this drop-down list you can select the action to be performed for incoming packets: • **Allow** incoming packets (default value). • **Block** incoming packets.
Always add allowing rules for Network Agent ports	This check box enables or disables the automatic adding of allowing rules for Network Agent ports. This check box is selected by default.

Network packet rules window

The **Network packet rules** table contains network packet rules that the Firewall Management component uses for network activity monitoring. You can configure the settings described in the table below for network packet rules.

Network packet rules settings

Setting	Description
Name	Network packet rule name.
Action	Action to be performed by Firewall Management when it detects the network activity.
Local address	Network addresses of devices that have Kaspersky Embedded Systems Security installed and can send and/or receive network packets.
Remote address	Network addresses of remote devices that can send and/or receive network packets.
Logging	This column shows if the application logs actions of the network packet rule. If the value is **Yes**, the application logs the actions of the network packet rule. If the value is **No**, the application does not log the actions of the network packet rule.

By default, the table of network packet rules is empty.

You can add, edit, delete, move up, and move down network packet rules in the table.

Added network packet rule window

In this window, you can configure the added network packet rule settings.

Network packet rule settings

Setting	Description
Protocol	You can select the type of data transfer protocol for which you want to monitor network activity: • **Any** (default value) • **GRE** • **ICMP** • **ICMPv6** • **IGMP** • **TCP** • **UDP**
Direction	You can specify the direction of network activity being monitored: • **Incoming packets**. If this option is selected, the Firewall Management component monitors only incoming packets. • **Incoming**. If this option is selected, the Firewall Management component monitors only incoming connections. • **Incoming/Outgoing** (default setting). If this option is selected, the Firewall Management component monitors both incoming and outgoing connections. • **Incoming/Outgoing packets**. If this option is selected, the Firewall Management component monitors both incoming and outgoing packets. • **Outgoing packets**. If this option is selected, the Firewall Management component monitors only outgoing packets. • **Outgoing**. If this option is selected, the Firewall Management component monitors only outgoing connections.
ICMP type	You can specify the ICMP type. The Firewall Management component monitors messages of the specified type sent by the host or gateway. If the **Specified** option is selected, the field for entering the ICMP type will be displayed. This window is displayed if the **ICMP** or **ICMPv6** data transfer protocol is selected in the **Protocol** drop-down list.
ICMP code	You can specify the ICMP code. The Firewall Management component monitors messages of the type specified in the **ICMP type** field, with the code specified in the **ICMP code** field, and sent by the host or gateway. If the **Specified** option is selected, the field for entering the ICMP code will be displayed. This window is displayed if the **ICMP** or **ICMPv6** data transfer protocol is selected in the **Protocol** drop-down list.
Remote ports	You can specify a port number or port range of the remote devices between which the connection is to be monitored. If the **Specified** option is selected, the field for entering the port numbers will be displayed. This window is displayed only if **TCP** or **UDP** data transfer protocol is selected in the **Protocol** drop-down list.
Local ports	You can specify a port number or port range of the local devices between which the connection is to be monitored. If the **Specified** option is selected, the field for entering the port numbers will be displayed. This window is displayed only if **TCP** or **UDP** data transfer protocol is selected in the **Protocol** drop-down list.

Setting	Description
Remote addresses	You can specify the network addresses of the remote devices that can send and receive network packets: • **Any address** (default value). If this item is selected, the network rule controls network packets sent and/or received by remote devices with any IP address. • **Specified address or address range**. If this option is selected, the network rule controls the sending and receiving of network packets by remote devices with the IP addresses that are specified in the table below. Clicking the **Add** button above the table opens the Add IP addresses window, in which you can specify an IP address of the device, a network address, or a range of IP addresses. • **By network type**. If this item is selected, the network rule controls network packets sent and received by remote devices with the IP addresses associated with the selected network type: **Public networks**, **Local networks**, or **Trusted networks**.
Local addresses	You can specify the network addresses of the devices with Kaspersky Embedded Systems Security installed that can send and receive network packets: • **Any address** (default value). If this option is selected, the network rule controls network packets sent and/or received by the devices with Kaspersky Embedded Systems Security installed regardless of their IP address. • **Specified address or address range**. If this option is selected, the network rule controls the network addresses of devices with Kaspersky Embedded Systems Security installed that can send and receive network packets. These network addresses are specified in the table below. Clicking the **Add** button above the table opens the Add IP addresses window, in which you can specify an IP address of the device, a network address, or a range of IP addresses.
Action	You can select an action to be performed by the Firewall Management component when it detects network activity: • **Block** network activity. • **Allow** network activity (default value).
Logging	You can specify whether the actions of the network rule will be logged in the report.
Rule name	The field for entering the name of the network packet rule.

Add IP addresses window

In this window, you can specify the IP address of the device, network address or range of IP addresses.

You can specify multiple addresses; enter each address on a new line for convenience of copying them.

Available networks window

The **Available networks** table contains the networks controlled by the Firewall Management component. The table of available networks is empty by default.

Available networks settings

Setting	Description
IP address	Network IP address.
Network type	Network type (**Public network**, **Local network**, or **Trusted network**).

You can add, edit, and delete available networks.

Network connection window

In this window, you can configure the network connection that the Firewall Management component will monitor.

Network connection

Setting	Description
IP address	The field for entering IP address of the network.
Network type	You can select the type of the network: • **Public** • **Local** • **Trusted**

Firewall Management in the command line

In the command line, you can configure Firewall Management using the Firewall Management predefined task (*Firewall_Management*).

By default, the Firewall Management Task is not run. You can start and stop this task manually.

You can configure the firewall management settings by editing the settings of the Firewall Management predefined task.

You can also configure Firewall Management settings using Firewall Management commands:

- Create and delete network packet rules and change their execution priority.
- Create a list of IP addresses or subnets in network zones.
- View firewall rules created in the application by using the following command: `kess-control -F --query`.

Firewall Management task settings

Setting	Description	Values
`DefaultIncomingAction`	The default action to perform on an inbound connection if no network rules apply to this connection type.	`Allow` (default value) — Allow inbound connections. `Block` — Block inbound connections.
`DefaultIncomingPacketAction`	The default action to perform on an incoming packet if no network packet rules apply to this connection type.	`Allow` (default value) — Allow incoming packets. `Block` — Block incoming packets.
`OpenNagentPorts`	Adds Network Agent dynamic rules to the network packet rules.	`Yes` (default value) – Add Network Agent dynamic rules to the network packet rules. `No` – Do not add Network Agent dynamic rules to the network packet rules.
The **[PacketRules.item_#]** section contains network packet rules for the Firewall Management task. You can specify several `[PacketRules.item_#]` sections in any order. The application processes the scopes by index in ascending order. Each `[PacketRules.item_#]` section contains the following settings:		
`Name`	Network packet rule name.	Default value: `Packet rule #<n>`, where n is an index.
`FirewallAction`	Action to be performed on connections specified in this network packet rule.	`Allow` (default value) — Allow network connections. `Block` — Block network connections.

Setting	Description	Values

`Protocol`	Type of protocol for which network activity is to be monitored.	`Any` (default value) — The Firewall Management task monitors all network activity.
		`TCP`
		`UDP`
		`ICMP`
		`ICMPv6`
		`IGMP`
		`GRE`

`RemotePorts`	Port numbers of the remote devices whose connection is monitored. An integer or interval can be specified for this value. This setting can only be specified if the `Protocol` setting is set to `TCP` or `UDP`.	`Any` (default value) — Monitor all remote ports. `0 – 65535`.
`LocalPorts`	Port numbers of the local devices whose connection is monitored. An integer or interval can be specified for this value. This setting can only be specified if the `Protocol` setting is set to `TCP` or `UDP`.	`Any` (default value) — Monitor all local ports. `0 – 65535`.
`ICMPType`	ICMP packet type. This setting can only be specified if the `Protocol` setting is set to `ICMP` or `ICMPv6`.	`Any` (default value) — Monitor all ICMP packet types. Integer number according to the data transfer protocol specification.
`ICMPCode`	ICMP packet code. This setting can only be specified if the `Protocol` setting is set to `ICMP` or `ICMPv6`.	`Any` (default value) — Monitor all ICMP packet codes. Integer number according to the data transfer protocol specification.
`Direction`	Direction of the monitored network activity.	`IncomingOutgoing` or `InOut` (default value) — Monitor both inbound and outbound connections. `Incoming` or `In` — Monitor only inbound connections. `Outgoing` or `Out` — Monitor only outbound connections. `IncomingPacket` or `InPacket` — Monitor only incoming packets. `OutgoingPacket` or `OutPacket` — Monitor only outgoing packets. `IncomingOutgoingPacket` or `InOutPacket` — Monitor both incoming and outgoing packets.
`RemoteAddress`	The network addresses of the remote devices that can send and receive network packets.	`Any` (default value) — Monitor network packets sent and/or received by remote devices with any IP address. `Trusted` — Predefined network zone for trusted networks. `Local` — Predefined network zone for local networks. `Public` — Predefined network zone for public networks. `d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d – d.d.d.d` — Range of IPv4 addresses. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x:x:x:x:x – x:x:x:x:x:x:x:x` — Range of IPv6 addresses. x:x:x:x:x:x:x:x/p — Subnet of IPv6 addresses, where p is a number from 0 to 128; you can use **::** for brevity.

Setting	Description	Values
`LocalAddress`	Network addresses of devices that have Kaspersky Embedded Systems Security installed and can send and/or receive network packets.	`Any` (default value) — Monitor network packets sent and/or received by local devices with any IP address. `d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d – d.d.d.d` — Range of IPv4 addresses. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x:x:x:x:x – x:x:x:x:x:x:x:x` — Range of IPv6 addresses. x:x:x:x:x:x:x:x/p — Subnet of IPv6 addresses, where p is a number from 0 to 128; you can use :: for brevity.
`LogAttempts`	Include a record of the network rule action in the report.	`Yes` — Log actions in the report. `No` (default value)—Do not write the actions in the report.
The **[NetworkZonesPublic]** section contains network addresses associated with public networks. You can specify several IP addresses or subnets of IP addresses.		
`Address.item_#`	Specifies IP addresses or subnets of IP addresses.	`d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x::0/p` — Subnet of IPv6 addresses, where p is a number from 0 to 64. Default value: "" (no network addresses in this zone)
The **[NetworkZonesLocal]** section contains network addresses associated with local networks. You can specify several IP addresses or subnets of IP addresses.		
`Address.item_#`	Specifies IP addresses or subnets of IP addresses.	`d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x::0/p` — Subnet of IPv6 addresses, where p is a number from 0 to 64. Default value: `""` (no network addresses in this zone)
The **[NetworkZonesTrusted]** section contains network addresses associated with trusted networks. You can specify several IP addresses or subnets of IP addresses.		
`Address.item_#`	Specifies IP addresses or subnets of IP addresses.	`d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x::0/p` — Subnet of IPv6 addresses, where p is a number from 0 to 64. Default value: `""` (no network addresses in this zone)

Configuring a list of network packet rules on the command line

To add a network packet rule, execute the following command:

```
kess-control --add-rule [--name <rule name>] [--action <action>] [--protocol
<protocol>] [--direction <direction>] [--remote <remote address>[:<port or port
range>]] [--local <local address>[:<port or port range>]] [--at <index>]
```

where:

- `--name <rule name>` is the name of the network packet rule.
- `--action <action>` is the action to be performed on connections specified in network packet rule.
- `--protocol <protocol>` is the type of communication protocol for which you want to monitor network activity.
- `--direction <direction>` is the direction of the monitored network activity.
- `--remote <remote address>[:<port or port range>]` is the network address of the remote device. You can specify the name of a predefined network zone as the remote address.
- `--local <local address>[:<port or port range>]` is the network address of the Kaspersky Embedded Systems Security device.
- `--at <index>` is the index of rules in the list of network packet rules. If the `--at` option is not specified or its value is larger than the number of rules in the list, the new rule is added to the end of the list.

Parameters that you do not specify values for in the command are set to their default values.

Examples:

To create a rule that blocks all incoming and established connections to TCP port 23, execute the following command:

```
kess-control --add-rule --name Block_Telnet --action Block --direction in --protocol TCP --local any:23 --remote
any
```

To create a rule that blocks incoming and established connections via the TCP port 23 for the Public network zone, execute the following command:

```
kess-control --add-rule --name Block_Telnet --action Block --direction in --protocol TCP --local any:23 --remote
Public
```

To delete a network packet rule, execute one of the following commands:

- `kess-control --del-rule --name <rule name>`
- `kess-control --del-rule --index <index>`

where:

- `--name <rule name>` is the name of the network packet rule.
- `--index <index>` is the current index of rules in the list of network packet rules.

If the list of network packet rules contains multiple rules with an identical name or does not contain a rule with a specified name or index, an error occurs.

To change a network packet rule's execution priority, execute one of the following commands:

- `kess-control --move-rule --name <rule name> --at <index>`
- `kess-control --move-rule --index <index> --at <index>`

  where:

- `--name <rule name>` is the name of the network packet rule.
- `--index <index>` is the current index of rules in the list of network packet rules.
- `--at <index>` is the new index of rules in the list of network packet rules.

Configuring network zones on the command line

To add a network address to the zone, execute the following command:

```
kess-control --add-zone <zone> --address <address>
```

where:

- `<zone>` is the predefined name of the network zone. Possible values: `Public`, `Local`, `Trusted`.
- `<address>` is the network address or subnet.

To delete a network address from a zone, execute one of the following commands:

- `kess-control --del-zone <zone> --address <address>`
- `kess-control --del-zone <zone> --index <address index in the zone>`

If a zone contains several items with the same network address, the `--del-zone` command will not be executed.

If the specified network address or index does not exist, an error message is generated.

Web Threat Protection

The Web Threat Protection component allows you to scan inbound traffic via HTTP, HTTPS, and FTP, websites, and IP addresses, prevent malicious files from being downloaded from the Internet, and block access to phishing, adware, and other malicious websites. Web Threat Protection uses port 9879 to intercept connections.

> Stopping or starting the application or the Web Threat Protection component terminates connections on intercepted ports, for example, HTTPS.

By default, the Web Threat Protection task is disabled. However, it is enabled automatically if local management of Web Threat Protection settings has been allowed on the device (a policy is not applied or the "lock" is not set in the policy properties) and one of the following executable browser files, including in snap format, has been detected on the system:

- chrome
- chromium
- chromium-browser
- firefox
- firefox-esr
- google-chrome
- opera
- yandex-browser

You can enable or disable Web Threat Protection, and also configure the protection settings:

- Select action that the application performs on a web resource where a dangerous object is detected.
- Configure a list of trusted web addresses. The application will not scan the contents of websites whose web addresses are included in this list.
- Select objects that the application will detect when scanning inbound traffic.
- Configure the encrypted connections scan to scan HTTPS traffic.

> To scan FTP traffic, control of all network ports must be configured in the settings for the network traffic scan.

When a website is opened, the application performs the following actions:

1. Checks the website security using the downloaded application databases.
2. Checks the website security using **heuristic analysis**, if enabled.

   During heuristic analysis, Kaspersky Embedded Systems Security analyzes the activity of applications in the operating system. Heuristic analysis can detect dangerous objects for which there are currently no records in Kaspersky Embedded Systems Security databases.
3. Checks the trustworthiness of a website using Kaspersky reputation databases if the use of Kaspersky Security Network is enabled.

   > You are advised to enable the use of Kaspersky Security Network to help Web Threat Protection work more effectively.
4. Blocks or allows opening of the website.

On attempt to open a dangerous website, the application performs the following:

- For HTTP or FTP traffic, the application blocks access and shows a warning message.
- For HTTPS traffic, a browser displays an error page.

> Removing application certificates may cause the Web Threat Protection component to work incorrectly.

> If you have configured traffic interception exclusions for Kaspersky Embedded Systems Security, these affect Web Threat Protection.

Configuring Web Threat Protection in the Web Console

In the Web Console, you can manage Web Threat Protection settings in the policy properties (**Application settings** → **Communication & Web Protection** → **Web Threat Protection**).

Web Threat Protection component settings

Setting	Description
Trusted web addresses	List of web addresses that the Web Threat Protection considers trusted. The component does not scan incoming and outgoing traffic from these addresses for threats. The list of trusted addresses can be used, for example, if the Web Threat Protection component interferes with downloading a file from a known website. You can add both the specific address and the address mask of a web page / website to the list of trusted web addresses. Kaspersky Embedded Systems Security supports `*` and `?` characters in **masks**. Masks are not supported to specify IP addresses. When creating an address mask, use an asterisk (*) as a placeholder for one or more characters. If you enter the *abc* address mask, it is applied to all web resources that contain the "abc" sequence (for example, www.virus.com/download_virus/page_0-9abcdef.html). To include the asterisk in the address mask as a character, but not as a mask, enter the * character twice (for example, www.virus.com/**/page_0-9abcdef.html means www.virus.com/*/page_0-9abcdef.html).
Action on threat detection	**Inform**. When this option is selected, if an infected object is detected in web traffic, the Web Threat Protection component informs the user about the detection of an infected object in web traffic and allows the object to be downloaded to the device. Web Threat Protection also logs information about the detection of the infected object and adds this information to the list of active threats. **Block**. When this option is selected, if an infected object is detected in web traffic, the Web Threat Protection component blocks access to the object, displays a message in the browser about the blocked access attempt, and logs information about the detection of an infected object.

Configuring Web Threat Protection in the Administration Console

In the Administration Console, you can manage Web Threat Protection settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → Web Threat Protection**).

Web Threat Protection component settings

Setting	Description
Enable Web Threat Protection	This check box enables or disables Web Threat Protection. This check box is cleared by default.
Trusted web addresses	This group of settings contains the **Configure** button, which opens the **Trusted web addresses** window, where you can specify the list of trusted web addresses. The application will not scan the contents of websites whose web addresses are included in this list.
Action on threat detection	Action that the application performs on a web resource where a dangerous object is detected: • **Block** access to all dangerous objects detected in web traffic, display a notification about the blocked access attempts, and log information about the dangerous objects (default value). • **Inform** the user when a dangerous object is detected in web traffic. Web Threat Protection allows this object to be downloaded to the device. At that, the application logs the information about the dangerous object and adds it to the list of active threats.
Scan settings	This group of settings contains the **Configure** button, which opens the **Scan settings** window, where you can configure the settings for scanning incoming traffic.

Trusted web addresses window

In this window, you can add web addresses and web pages whose content you consider trusted.

You can only add HTTP/HTTPS web addresses to the list of trusted web addresses. You can use **masks** to specify web addresses. Masks are not supported to specify IP addresses. By default, the list is empty. You can add, edit, and remove web addresses on the list.

When creating an address mask, use an asterisk (*) as a placeholder for one or more characters. If you enter the *abc* address mask, it is applied to all web resources that contain the "abc" sequence (for example, www.virus.com/download_virus/page_0-9abcdef.html). To include the asterisk in the address mask as a character, but not as a mask, enter the * character twice (for example, www.virus.com/**/page_0-9abcdef.html means www.virus.com/*/page_0-9abcdef.html).

Scan settings window

In this window, you can configure the settings for scanning incoming traffic during operation of the Web Threat Protection component.

Web Threat Protection settings

Setting	Description
Detect malicious objects	This check box enables or disables checking of links against the databases of malicious web addresses. This check box is selected by default.
Detect phishing links	This check box enables or disables checking of links against the databases of phishing web addresses. This check box is selected by default.
Use heuristic analysis for detecting phishing links	This check box enables or disables the use of heuristic analysis for detecting phishing links. This check box is available if the **Detect phishing links** check box is selected, and is selected by default.
Detect adware	This check box enables or disables checking links against the databases of adware web addresses. This check box is cleared by default.
Detect legitimate applications that intruders can use to compromise devices or data	This check box enables or disables checking links against the databases of legitimate applications that intruders can use to compromise devices or data. This check box is cleared by default.

Configuring Web Threat Protection on the command line

In the command line, you can manage Web Threat Protection using the Web Threat Protection predefined task (*Web_Threat_Protection*).

The task starts automatically if one of the supported browsers is detected in the system and local management of Web Threat Protection settings is allowed on the device (a policy is not applied or the "lock" is not set in the policy properties). You can start and stop the task manually.

You can configure Web Threat Protection settings by editing the settings of the Web Threat Protection predefined task.

Web Threat Protection task settings

Setting	Description	Values
`ActionOnDetect`	Specifies the action to be performed upon detection of an infected object in web traffic.	`Notify` — Allow the detected object to be downloaded, display a notification about the blocked access attempt, and log information about the infected object. `Block` (default value) — Block access to the detected object, display a notification about the blocked access attempt, and log information about the infected object.
`CheckMalicious`	Enables or disables checking of links against the databases of malicious web addresses.	`Yes` (default value) — Check if the links are listed in the malicious links database. `No` — Do not check if the links are listed in the malicious links database.
`CheckPhishing`	Enables or disables checking of links against the databases of phishing web addresses.	`Yes` (default value) — Check if the links are listed in the phishing links database. `No` — Do not check if the links are listed in the phishing links database.
`UseHeuristicForPhishing`	Enables or disables the use of heuristic analysis for scanning web pages for phishing links.	`Yes` (default value) — Use heuristic analysis to detect phishing links. If this value is specified, the level of heuristic analysis is `Light` (the least thorough scan with minimal load on the system). You cannot change the heuristic analysis level for the Web Threat Protection task. `No` — Do not use heuristic analysis to detect phishing links.
`CheckAdware`	Enables or disables checking of links against the databases of adware web addresses.	`Yes` — Check if the links are listed in the adware links database. `No` (default value) — Do not check if the links are listed in the adware links database.
`CheckOther`	Enables or disables the scanning of links against the database of web addresses containing legitimate applications that intruders can use to compromise the devices or data.	`Yes` —Check if the links are listed in the database of web addresses that contain legal applications that may be used by intruders to damage your devices or data. `No` (default value) — Do not check if the links are listed in the database of web addresses that contain legal applications that may be used by intruders to damage your devices or data.
`UseTrustedAddresses`	Enables or disables the usage of a list of trusted web addresses. The application does not scan trusted web addresses for viruses or other malicious objects. You can specify trusted web addresses using the `TrustedAddresses.item_#` setting.	`Yes` (default value) — Use a list of trusted web addresses. `No` — Do not use a list of trusted web addresses.
`TrustedAddresses.item_#`	Specifies trusted web addresses.	The default value is not defined. You can use **masks** to specify web addresses. When creating an address mask, use an asterisk (*) as a placeholder for one or more characters. If you enter the *abc* address mask, it is applied to all web resources that contain the "abc" sequence (for example, www.virus.com/download_virus/page_0-9abcdef.html). To include the asterisk in the address mask as a character, but not as a mask, enter the * character twice (for example, www.virus.com/**/page_0-9abcdef.html means www.virus.com/*/page_0-9abcdef.html). Masks are not supported to specify IP addresses.

Scanning network traffic

Network traffic scanning settings are used by Web Threat Protection and Network Threat Protection. The Web Threat Protection component can decrypt and inspect network traffic. The encrypted connections scan is enabled by default.

You can manage the following network traffic scan settings:

- Enable or disable encrypted connections scanning.
- Select the action to be performed by the application upon detection of an untrusted certificate.
- Select the action to be performed when a network traffic scan error occurs on a website.
- Enable or disable the use of the Internet for certificate verification.
- View and configure a list of trusted domains. The application will not scan connections established when visiting specified domains.
- Configure a list of root certificates that the application will consider trusted when performing an encrypted connections scan.
- Configure traffic interception exclusions You can specify connections to be excluded from traffic interception. An excluded connection is defined by the following parameters:
  - Traffic direction, inbound or outbound.
  - Remote IP address, the IP address to which the traffic excluded from interception is going to or coming from (depending on the direction of traffic).
  - Destination port, a port on a protected device (in an inbound traffic interception exclusion) or a port on a remote device (in an outbound traffic interception exclusion).

  Traffic interception exclusions are applied only to TCP connections and do not affect network packet rules. Traffic interception exclusions affect Web Threat Protection and Network Threat Protection protection components.

  You can also configure traffic interception exclusions for Kaspersky Embedded Systems Security using operating system tools.
- Configure a list of network ports to be monitored by the application. You can specify the network ports or network port ranges to be monitored.

  If you want to intercept network connections to the NFS server, you need to make sure that the NFS server is started with the `insecure` option (so that it accepts connections from ports outside the 1–1024 range), otherwise the NFS server may reject such connections.
- Select the traffic interception mode that the application uses.

When the network traffic scan settings are changed, the application records a *NetworkSettingsChanged* event in the log file.

About network traffic interception modes

An application can work in one of two network traffic interception modes:

- *Use eBPF*. If this mode is selected, the application uses the eBPF technology to intercept traffic.
- *Use iptables*. If this mode is selected, the application uses the iptables technology to intercept traffic. This mode is selected by default.

In the **Use eBPF** mode, the application creates its own separate network namespace (network linux namespace), where it redirects network packets using eBPF programs associated with `tc` (linux traffic control) queues of local network interfaces. The redirected network traffic is then analyzed in accordance with active network tasks and then is returned to the original network interface.

For communication between namespaces, pairs of `veth` virtual network interfaces are used (each side of a pair is in its own network namespace):

- The kess_in – kess_worm_in pair transmits network traffic of inbound connections.
- The kess_out – kess_worm_out pair transmits network traffic of outbound connections.

Through these pairs of interfaces, network traffic is sent for scanning from the default namespace to the namespace created by the application, and through the same interfaces, the traffic is returned back after analysis.

Network traffic reaches these interfaces from other network interfaces in the system only if the application is running and network interception is enabled. These interfaces cannot be used for data transmission over the network between remote devices.

If your device is sensitive to network port enumeration attacks (for example, if the operating system firewall with port filtering is configured on the local device), then we do not recommend selecting the **Use eBPF** mode. We also do not recommend using this traffic interception mode on devices of the organization's network perimeter.

If the device is located in an isolated network segment, in a local network, or inside the perimeter of the organization's network, the traffic interception mode **Use eBPF** mode may be selected.

You can select the traffic interception mode in the Web Console, the Administration Console, or on the command line.

To change the traffic interception mode, you need to restart the application. Restarting breaks established network connections.

Configuring network traffic scanning in the Web Console

In the Web Console, you can manage network traffic scan settings in the policy properties (**Application settings** → **General Settings** → **Network Settings** and **Application settings** → **Trusted Zone**).

Encrypted connections scan settings

Setting	Description
Proxy server	Proxy server settings for internet access of users of client computers. Kaspersky Embedded Systems Security uses these settings for certain protection components, including the application database and module update. The proxy server is not used by default.
Proxy server exclusions	This group of settings contains the **Settings** button, which you can click to open the **Proxy server exclusions** window, in which you can disable the proxy server for connections to the specified addresses. The table contains a list of IP addresses and DNS names for which the proxy server must not be used. You can use masks (`*` and `?` characters) to specify addresses.

Encrypted connections scan	The encrypted connections scan is enabled by default. Clicking the **Settings** button opens the **Network Settings** window, in which you can manage the following settings:
	• **Visiting a domain with an untrusted certificate** specifies the action that the application must perform when a domain with an untrusted certificate is visited:
	  • **Allow connection** (default value) allows connecting to the domain with an untrusted certificate.
	  • **Block connection** blocks the connection to the domain with an untrusted certificate.
	• **Visiting a domain with an encrypted connections scan error** specifies the action that the application must perform when a domain with a network traffic scan error is visited:
	  • **Allow and add domain to exclusions** (default value) adds the domain that caused an error to the list of domains with scan errors and does not scan encrypted network traffic when this domain is visited.
	  • **Block connection** blocks the connection to the domain with a scan error.
	• **Certificate verification policy** is the method the application uses to verify certificates:
	  • **Local check**: the application does not use the internet to verify a certificate.
	  • **Full check** (default value): the application uses the internet to check and download the missing chains that are needed to validate a certificate.

Monitored ports	Network ports monitoring mode: • **Monitor all network ports**. Network port monitoring mode in which the application monitors the data streams passing through any open network ports of the device. • **Monitor selected network ports only** (default). Network port monitoring mode in which the application monitors the selected network ports of the device. Clicking the **Monitored ports** link opens the **Monitored ports** window, where you can specify the network ports to be monitored by the application. The list of network ports that are normally used for transmission of email and network traffic is configured according to the recommendations of Kaspersky experts.
Traffic interception mode:	Traffic interception mode: **eBPF**. Traffic interception mode in which the application uses the eBPF technology to intercept traffic. **iptables** (default value). Traffic interception mode in which the application uses the iptables utility to intercept traffic. To change the traffic interception mode, you need to restart the application. Restarting breaks established network connections.

Configuring network traffic scanning in the Administration Console

In the Administration Console, you can manage network traffic scan settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls → Network settings**).

Encrypted connections scan settings

Setting	Description
Enable encrypted connection scans	This check box enables or disables encrypted connection scans. This check box is selected by default.

Settings for encrypted connections	This section contains the encrypted connections scan settings:
	• **Visit to a domain with an untrusted certificate** specifies the action that the application must perform when a domain with an untrusted certificate is visited:
	  • **Allow** (default value) — Allow connecting to the domain with an untrusted certificate.
	  • **Block** — Block connection to the domain with an untrusted certificate.
	• **Visit to a domain with a network traffic scan error** specifies the action that the application must perform when a domain with a network traffic scan error is visited:
	  • **Allow and add domain to exclusions** (default value) — Add the domain that resulted in the error to the list of domains with scan errors and do not scan encrypted network traffic when this domain is visited.
	  • **Block** — Block connection to the domain with a scan error.
	• **Certificate verification policy** is the method the application uses to verify certificates:
	  • **Local check**: the application does not use the internet to verify a certificate.
	  • **Full check** (default value): the application uses the internet to check and download the missing chains that are needed to verify a certificate.

Trusted domains	This group of settings contains the **Configure** button, which opens the Trusted domains window, where you can configure the list of trusted domain names.
Trusted root certificates	This group of settings contains the **Configure** button. Clicking this button opens the Trusted root certificates window. In this window, you can configure the list of root certificates that Kaspersky Embedded Systems Security must consider trusted when scanning encrypted connections.
Traffic interception exclusions	This group of settings contains the **Configure** button, which you can click to open the Traffic interception exclusions window. In this window, you can manage the settings of connections that the application excludes from traffic interception.
Network ports settings	This group of settings contains the **Configure** button. Clicking this button opens the Monitored ports window.
Advanced settings	This group of settings contains the **Configure** button, which you can click to open the **Traffic interception mode** window. In this window, you can configure the traffic interception mode: • **Use eBPF**. If this option is selected, the application uses the eBPF technology to intercept traffic. • **Use iptables** (default). If this option is selected, the application uses the iptables technology to intercept traffic. To change the traffic interception mode, you need to restart the application. Restarting breaks established network connections.

Trusted domains window

This list contains the domain names and domain name masks that will be excluded from scanning. The application does not scan connections established when visiting specified domains.

Example: `*example.com`. For example, `*example.com/*` is incorrect because a domain address, not a web page, needs to be specified.

By default, the list is empty. You can add, edit, or remove domains from the list of trusted domains.

Trusted certificates window

You can configure a list of root certificates considered trusted by Kaspersky Embedded Systems Security. The list of trusted root certificates is used when scanning encrypted connections.

The following information is displayed for each certificate:

- Certificate subject
- Certificate serial number
- Certificate issuer
- Certificate start date
- Certificate expiration date
- SHA256 certificate fingerprint

By default, the certificate list is empty.

You can add or remove certificates.

Clicking the **Add** button opens a window in which you can add a certificate to the list of trusted certificates in one of the following ways:

- Indicate the path to the certificate file. The **Browse** button opens the standard file selection window. Indicate the path to the file that contains the certificate, in DER or PEM format.
- Copy the contents of the certificate file to the **Enter certificate details** field.

Traffic interception exclusions window

The table contains the settings of connections that the application excludes from traffic interception:

- **Remote IP address** is the IP address to which the traffic excluded from interception is going to or from which it is coming from (depending on the direction of traffic).
- **Destination port** is a port on a protected device (in an inbound traffic interception exclusion) or a port on a remote device (in an outbound traffic interception exclusion).
- **Traffic direction** is the direction of traffic in the connection: inbound or outbound.

You can add, edit, or delete exclusions.

Traffic interception exclusion window

In this window, you can add a traffic interception exclusion. To do this, you need to specify the following connection settings:

- **Remote IP address** is the IPv4 or IPv6 address of the remote device. Depending on the traffic direction, traffic coming from the specified IP address (exclusion for inbound traffic) or traffic going to the specified IP address (exclusion for outgoing traffic) is excluded.
- **Destination port**, depending on traffic direction:
  - Port on the protected device (exclusion for inbound traffic). Traffic to this port from remote devices is excluded from interception.
  - Port on the remote device (exclusion for outbound traffic). Traffic to this port from the protected device is excluded from interception.
- **Traffic direction** is the direction of traffic in the excluded connection. You can configure an exclusion for inbound traffic or for outbound traffic.

Monitored ports

Network ports settings

Setting	Description
Monitor all network ports	If this option is selected, the application monitors all network ports.
Monitor selected network ports only	If this option is selected, the application monitors only the network ports specified in the table. This option is selected by default.
Network ports settings	This table contains network ports monitored by the application if the **Monitor selected network ports only** option is selected. The table contains two columns: • **Port** – monitored port. • **Description** – description of the monitored port. By default, the table displays a list of network ports that are usually used for the transmission of mail and network traffic. The list of network ports is included in the application package. You can add, edit, and delete items in the table.

Configuring network traffic scanning on the command line

On the command line, you can use special management commands to manage network traffic scan settings. You can use network traffic scan settings management commands to:

- Manage network traffic scan settings.
- View exclusions from the encrypted connections scan.
- Clear the list of domains that the application automatically excluded from the scan.

- Configure the list of trusted root certificates that the application uses when scanning encrypted connections.
- Configure and view traffic interception exclusions.

Viewing and editing network traffic scan settings

You can use special network traffic scan settings management commands to:

- Output the current values of network traffic scan settings to the console or to a configuration file. You can use this file to edit the settings.
- Manage all network traffic scan settings using the configuration file. You can get a configuration file that you can edit using the display the network traffic scan settings output command.
- Edit individual settings using command line options in the format `<setting name>=<setting value>`. You can get the current values of the settings using the command to output network traffic scan settings.

To output the current network traffic scan settings to the console, execute the following command:

```
kess-control --get-net-settings [--json]
```

where `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

To output the current network traffic scan settings to a file, run the following command:

```
kess-control --get-net-settings --file <path to configuration file> [--json]
```

where:

- `--file <configuration file path>` is the path to the configuration file to which you want to save the network traffic scan settings. If you specify the name of a file without specifying its path, the file will be created in the current directory. If a file with the specified name already exists in the specified path, it will be overwritten. If the specified directory cannot be found on the disk, file will not be created.
- `--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

To manage network scan settings using a configuration file:

1. Output the network traffic scan settings to the configuration file, as described above.
2. Edit the values of the necessary parameters in the file and save the changes.
3. Execute the command:

   ```
   kess-control --set-net-settings --file <path to configuration file> [--json]
   ```

where:

- `--file <configuration file path>` is the full path to the configuration file with network traffic scan settings.
- `--json` is specified to import the settings from the configuration file into the application in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

All network traffic scan settings configured in the file are imported into the application.

To edit network traffic scan settings on the command line, run the following command:

```
kess-control --set-net-settings <setting name>=<setting value> [<setting name>=
<setting value>]
```

where `<setting name>=<setting value>` is the name and value of one of the network traffic scan settings.

The values of the specified settings are modified.

Viewing exclusions from encrypted connections scan

You can view the following lists of exclusions from the encrypted connections scan:

- a list of exclusions added by the user;
- a list of exclusions added by the application;
- list of exclusions received from the application databases.

To view the list of encrypted connection scan exclusions added by a user, execute the following command:

```
kess-control -N --query user
```

To view the list of encrypted connection scan exclusions added by a user, execute the following command:

```
kess-control -N --query auto
```

To view the list of encrypted connection scan exclusions received from the application databases, execute the following command:

```
kess-control -N --query kl
```

To clear a list of domains that the application automatically excluded from scan, execute the following command:

```
kess-control [-N] --clear-web-auto-excluded
```

Managing the list of trusted root certificates

To add a certificate to the list of trusted root certificates, run the following command:

```
kess-control --add-certificate <path to certificate>
```

where:

`<path to certificate>` is the path to the certificate file that you want to add (PEM or DER format).

To remove a certificate from the list of trusted root certificates, run the following command:

```
kess-control --remove-certificate <certificate subject>
```

To view the list of trusted root certificates, execute the following command:

```
kess-control --list-certificates
```

The following information is displayed for each certificate:

- Certificate subject
- serial number
- Certificate issuer
- Certificate start date
- Certificate expiration date
- SHA256 certificate fingerprint

Configuring and viewing traffic interception exclusions

To add a traffic interception exclusion, run the following command:

```
kess-control --add-bypass-endpoints --direction <in|out> --remote-ip <remote IP
address> --dst-port <destination port>
```

where:

- `<in|out>` is the direction of traffic in the excluded connection. You can configure an exclusion for inbound traffic or for outbound traffic.
- `<remote IP address>` is the IPv4 or IPv6 address of the remote device. Depending on the traffic direction, traffic coming from the specified IP address (exclusion for inbound traffic) or traffic going to the specified IP address (exclusion for outgoing traffic) is excluded.

- < destination port >, depending on traffic direction:
  - Port on the protected device (exclusion for inbound traffic). Traffic to this port from remote devices is excluded from interception.
  - Port on the remote device (exclusion for outbound traffic). Traffic to this port from the protected device is excluded from interception.

In an exclusion, you may specify an IP address of the remote device without a port or a destination port without an IP address.

If you specify only the IP address of the remote device in an exclusion for inbound traffic (without specifying a port on the protected device), all traffic from the remote device is excluded from interception. We recommend configuring this exclusion only for traffic from devices that you trust.

To view the list of configured traffic interception exclusions, run the following command:

```
kess-control --list-bypass-endpoints
```

The following information is displayed for each exclusion:

- Serial number of the exclusion
- Direction of traffic
- IP address of the remote device
- Port number

To delete a traffic interception exclusion, run the following command:

```
kess-control --remove-bypass-endpoints --direction <in|out> --remote-ip < remote IP address > --dst-port < destination port >
```

where:

- <in|out> is the direction of traffic in the excluded connection.
- < remote IP address > is the IPv4 or IPv6 address of the remote device whose traffic is excluded from interception (depending on the direction of traffic).
- < destination port > is a port on the protected device (exclusion for inbound traffic) or a port on the remote device (exclusion for outbound traffic).

Exclusions from traffic interception using the tools of the operating system

In Kaspersky Embedded Systems Security settings, you can configure exclusions from traffic interception by traffic direction, destination IP address and port. If such exclusions are not sufficient to resolve the conflict of the Kaspersky Embedded Systems Security application with client applications, you can use the special chain of allow rules, kess_bypass.

The kesl_bypass chain is a special list of rules that Kaspersky Embedded Systems Security creates in the mangle table of the operating system firewall (iptables/ip6tables). The rules of the kess_bypass chain make it possible to exclude traffic from interception by Kaspersky Embedded Systems Security. The kess_bypass chain is added to the top of the network packet processing process, which means that kess_bypass rules are processed before all other iptables rules.

If the application uses the eBPF technology to intercept traffic, kess_bypass rules are not applied.

In contrast to exclusions that you can connect in Kaspersky Embedded Systems Security settings, the kess_bypass chain allows configuring exclusions not only by traffic direction, destination IP address and port, but also by other parameters of connections.

The rules in the kess_bypass chain can be changed by means of the operating system.

For example, you can add a rule to the kess_bypass chain by running the following command: `iptables -t mangle -I kess_bypass -m tcp -p tcp --dport http -j ACCEPT`.

This rule allows excluding inbound and outbound HTTP traffic on port 80 at any IP address from interception by Kaspersky Embedded Systems Security. This can help avoid potential conflict with web services that you trust.

If traffic exclusion rules are configured in the chain, these rules affect the Web Threat Protection and Network Threat Protection components.

When the application is removed, the kess_bypass rule chain in iptables and ip6tables is removed only if it was empty.

Network Threat Protection

The Network Threat Protection component allows you to scan inbound network traffic for activity that is typical for network attacks.

Kaspersky Embedded Systems Security receives TCP port numbers from the current application databases and scans incoming traffic for these ports. Network Threat Protection uses port 9789 to intercept connections.

To scan network traffic, the Network Threat Protection component receives port numbers from the application databases and accepts connections via all these ports. During the network scan process, it may look like an open port on the device, even if no application on the system is listening to this port. It is recommended to close unused ports by means of a firewall.

Current connections for intercepted TCP ports are reset when Network Threat Protection is enabled.

If Network Threat Protection is enabled, upon detecting an attempted network attack on a protected device, the application blocks network activity from the attacking device and creates the *Network attack detected* event. The event contains information about the attacking device. By default, network traffic from the attacking device is blocked for one hour. Once the blocking time has expired, the application unblocks the device.

By default, Network Threat Protection is disabled.

You can enable or disable Network Threat Protection, and also configure the protection settings:

- Select the action that the application will perform upon detection of network activity that is typical of network attacks.
- Enables or disables the blocking of network activity when a network attack attempt is detected.
- Set the duration for blocking an attacking device.
- Configure a list of IP addresses whose network activity will not be blocked by the application.

You can use the commands for administering blocked devices in the command line to view the list of blocked devices and manually unblock these devices. Kaspersky Security Center does not provide tools for monitoring and managing blocked devices, except for the *Network attack detected* events.

If you have configured traffic interception exclusions for Kaspersky Embedded Systems Security, these affect Network Threat Protection.

Configuring Network Threat Protection in the Web Console

In the Web Console, you can configure Network Threat Protection settings in the policy properties (**Application settings** → **Communication & Web Protection** → **Network Threat Protection**).

Network Threat Protection component settings

Setting	Description
Trusted IP addresses	List of IP addresses of trusted devices. The Network Threat Protection component does not block network activity from these addresses. Only an IPv4 or IPv6 address can be added to the exclusion list. The application also does not log information on network attacks from the IP addresses that are in the list of exclusions.
Action on threat detection	**Block**. When a threat is detected, the Network Threat Protection component blocks the attack. **Block attack and attacking device for (min)**. When a threat is detected, the Network Threat Protection component adds the attacking computer to the block list. This means that after the first attempt at a network attack, the Network Threat Protection component blocks the network connection with the attacking computer for the specified duration to automatically protect the user device from possible future network attacks from this address. The minimum time an attacking computer must spend in the block list is one minute. The maximum duration is 9999 minutes.

Configuring Network Threat Protection in the Administration Console

In the Administration Console, you can manage Network Threat Protection settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → Network Threat Protection**).

Network Threat Protection component settings

Setting	Description
Enable Network Threat Protection	This check box enables or disables Network Threat Protection. This check box is cleared by default.
Action on threat detection	Actions performed upon detection of network activity that is typical of network attacks. • **Inform** user. The application allows network activity and logs information about detected network activity. • **Block** network activity from an attacking device and log information about detected network activity (default value).
Block attacking devices	This check box enables or disables the blocking of network activity when a network attack attempt is detected. This check box is selected by default.
Block the attacking device for (min)	In this field you can specify the duration an attacking device is blocked in minutes. After the specified time, Kaspersky Embedded Systems Security allows network activity from this device. Possible values: integers from 1 to 32768. Default value: 60.
Exclusions	This group of settings contains the **Configure** button, which opens the **Exclusions** window, where you can specify a list of IP addresses. Network attacks from these IP addresses will not be blocked.

Exclusions window

In this window, you can add IP addresses from which network attacks will not be blocked.

By default, the list is empty.

You can add, edit, and remove IP addresses in the list. You can specify IP addresses in IPv4 and IPv6 formats.

Configuring Network Threat Protection on the command line

In the command line, you can manage Network Threat Protection using the Network Threat Protection predefined task (*Network_Threat_Protection*).

By default, the Network Threat Protection task does not run. You can start and stop the task manually.

You can configure Network Threat Protection settings by editing the settings of the Network Threat Protection predefined task.

Network Threat Protection task settings

Setting	Description	Values
`ActionOnDetect`	Action performed when network activity typical of network attacks is detected. Changing the value of this setting from `Block` to `Notify` clears the list of blocked devices.	`Notify` – allow network activity, log information about detected network activity. If this value is specified, the value of the `BlockAttackingHosts` parameter is ignored. `Block` (default value) – block network activity and log information about it.
`BlockAttackingHosts`	Blocking network activity from attacking devices.	`Yes` (default value) — Block network activity of an attacking device. `No` – Do not block network activity of the attacking device. If this value is specified and the `ActionOnDetect` parameter is set to `Block`, the application blocks network activity from the attacking device, but does not add the device to the list of blocked devices.
`BlockDurationMinutes`	Specifies how long attacking devices will be blocked (in minutes).	1 to 32768 Default value: 60.
`UseExcludeIPs`	The usage of a list of IP addresses whose network activity will not be blocked when a network attack is detected. The application will only log information about dangerous activity from these devices. You can add IP addresses to the exclusion list by using the `ExcludeIPs.item_#` setting.	`Yes` — Use the list of excluded IP addresses. `No` (default value) — Do not use the list of excluded IP addresses.
`ExcludeIPs.item_#`	Specifies an IP address whose network activity will not be blocked by the application. By default, the list is empty.	`d.d.d.d` — IPv4 address, where d is a decimal number from 0 to 255. `d.d.d.d/p` — Subnet of IPv4 addresses, where p is a number from 0 to 32. `x:x:x:x:x:x:x:x` — IPv6 address, where x is a hexadecimal number from 0 to ffff. `x:x:x:x::0/p` — Subnet of IPv6 addresses, where p is a number from 0 to 64. The default value is not defined.

BadUSB Attack Prevention

Some viruses modify the firmware of USB devices to trick the operating system into detecting the USB device as a keyboard. As a result, the virus can execute commands under your user account, for example, to download malware.

The BadUSB Attack Prevention protection component allows preventing the connection of infected USB devices that imitate a keyboard to the device.

When a USB device identified by the operating system as a keyboard is connected to the device, the application prompts the user to enter a numeric code generated by the application from the newly connected keyboard or using the on-screen keyboard (if available). This procedure is known as keyboard authorization.

If the code is entered correctly, the application saves the identification parameters, that is, the VID/PID of the keyboard and the port number to which it is connected, in the list of authorized keyboards. Keyboard authorization does not need to be repeated when the keyboard is reconnected or after the operating system is restarted.

> When an authorized keyboard is connected to a different USB port of the device, the application requires it to be authorized again.

If the numeric code is entered incorrectly, the application generates a new code. You can configure the number of attempts to enter the authorization code. If the numeric code is entered incorrectly multiple times or the keyboard authorization window is closed, the application blocks input from this keyboard. When the USB device blocking time elapses or the operating system is restarted, the application prompts the user to perform keyboard authorization again.

The application allows use of an authorized keyboard and blocks a keyboard that has not been authorized.

By default, the BadUSB Attack Prevention protection component is disabled. You can enable, disable, or configure BadUSB Attack Prevention using the Web Console, Administration Console, and the command line.

Configuring BadUSB Attack Prevention in the Web Console

In the Web Console, you can manage BadUSB Attack Prevention settings in the policy properties (**Application settings → Host Threat Protection → BadUSB Attack Prevention)**.

BadUSB Attack Prevention component settings

Setting	Description
Maximum USB device authorization attempts:	Automatically block the USB device if the authorization code is entered incorrectly the specified number of times. Valid values are 1 to 10. For example, if you allow five attempts to enter the authorization code, the application blocks the USB device after the fifth unsuccessful attempt. Kaspersky Embedded Systems Security displays the time for which the USB device is blocked. After the specified time has elapsed, you can try to enter the authorization code five more times.
Timeout when reaching the maximum number of attempts (min)	USB device block duration after the specified number of unsuccessful attempts to enter the authorization code. Possible values: 1 to 180 (minutes)
Prohibit use of On-Screen Keyboard for authorization of USB devices	If this toggle switch is enabled, the application does not allow to use the on-screen keyboard for keyboard authorization of a USB device that cannot be used to enter an authorization code.

Configuring BadUSB Attack Prevention in the Administration Console

In the Administration Console, you can manage Web Threat Protection settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Essential Threat Protection → BadUSB Attack Prevention**).

BadUSB Attack Prevention component settings

Setting	Description
Enable BadUSB Attack Prevention	This check box enables or disables the BadUSB Attack Prevention component. The component is disabled by default.
Maximum USB device authorization attempts:	Automatically block the USB device if the authorization code is entered incorrectly the specified number of times. Valid values are 1 to 10. For example, if you allow five attempts to enter the authorization code, the application blocks the USB device after the fifth unsuccessful attempt. Kaspersky Embedded Systems Security displays the time for which the USB device is blocked. After this time elapses, you will once again have five attempts to enter the authorization code. Default value: 5.
Blocking duration when the maximum number of authorization attempts is reached (minutes):	USB device block duration after the specified number of unsuccessful attempts to enter the authorization code. Possible values: 1 to 180 (minutes) Default value: 5.
Do not allow using an on-screen keyboard when authorizing USB devices	If this check box is selected, the application does not allow to use the on-screen keyboard for keyboard authorization of a USB device that cannot be used to enter an authorization code. This check box is cleared by default.

Configuring BadUSB Attack Prevention on the command line

On the command line, you can also manage BadUSB Attack Prevention using the BadUSB Attack Prevention predefined task (*Bad_USB*).

The BadUSB Attack Prevention task is not started by default. You can start and stop this task manually.

Administrator role privileges are required to start and stop the BadUSB Attack Prevention task from the command line.

You can manage BadUSB Attack Prevention settings by editing the settings of the BadUSB Attack Prevention predefined task.

The following table describes available values and default values of settings that you can specify for the BadUSB Attack Prevention task.

BadUSB Attack Prevention task settings

Setting	Description	Values
`MaxAuthAttempts`	Maximum number of USB device authorization attempts after which the application blocks the USB device and further attempts to authorize the USB device for the duration specified in the `TimeoutAfterAttempts` parameter.	Possible values: 1 to 10 Default value: 5.
`TimeoutAfterAttempts`	USB device blocking duration after `MaxAuthAttempts` unsuccessful attempts to enter an authorization code.	Possible values: 1 to 180 minutes. Default value: 5 minutes.
`ProhibitOnScreenKeyboard`	Prohibit or allow the on-screen keyboard for authorizing USB devices.	If `Yes`, the application does not allow using the on-screen keyboard for keyboard authorization of a USB device that cannot be used to enter an authorization code. If `No`, the application allows using the on-screen keyboard for keyboard authorization of a USB device that cannot be used to enter an authorization code.

Protection against remote malicious encryption

Anti-Cryptor component allows you to protect your files in local directories with network access by SMB/NFS protocols from remote malicious encryption.

If Anti-Cryptor is enabled, Kaspersky Embedded Systems Security scans the actions of remote devices with file resources located in shared network directories of the protected device for the presence of malicious encryption. If an application considers the actions of a remote device accessing shared network resources to be malicious encryption, the application creates and enables a rule for the firewall of the operating system that blocks network traffic from the compromised device. The compromised device is added to the list of untrusted devices, and access to shared network directories is blocked for all untrusted devices. The application creates an *Encryption detected* event that contains information about the compromised device.

By default, the application blocks access of untrusted devices to network file resources for 30 minutes. When the blocking time expires, the application deletes the compromised device from the list of untrusted devices, and the device's access to network file resources is automatically restored.

Firewall rules created by the Anti-Cryptor component cannot be deleted using the iptables utility, since the application restores a set of rules every minute.

Protection against remote malicious encryption is disabled by default.

You can enable or disable protection against malicious encryption (Anti-Cryptor), and also configure the protection settings:

- Select the action that the application will perform when encryption is detected: notify the user or block the device performing the malicious encryption.

  If the *Inform* action is selected, the application still scans remote devices' actions on network file shares to check for malicious encryption when Anti-Cryptor is enabled. If malicious activity is detected, the *Encryption detected* event is created, but the compromised device is not blocked.

- Set the duration for blocking an untrusted device.

- Specify the files and directories that the application protects against malicious encryption.

- Specify the files and directories that are excluded from protection against malicious encryption.

  The application does not consider actions to be encryption if encryption activity is detected in directories excluded from protection against encryption (Anti-Cryptor).

You can use the commands for administering blocked devices in the command line to view the list of blocked devices and manually unblock these devices. Kaspersky Security Center does not provide tools for monitoring and managing blocked devices, except for the *Encryption detected* events.

For the Anti-Cryptor component to operate correctly, at least one of the services (Samba or NFS) must be installed in the operating system. The NFS service requires the rpcbind package to be installed.

The Anti-Cryptor component runs correctly with SMB1, SMB2, SMB3, NFS3, TCP/UDP, and IP/IPv6 protocols. Working with NFS2 and NFS4 protocols is not supported. It is recommended to configure your server settings so that the NFS2 and NFS4 protocols cannot be used to mount resources.

Kaspersky Embedded Systems Security does not block access to network file resources until the device's activity is identified as malicious. So, at least one file will be encrypted before the application detects malicious activity.

Configuring Anti-Cryptor in the Web Console

In the Web Console, you can manage Anti-Cryptor settings in the policy properties (**Application settings** → **Host Threat Protection** → **Behavior Analysis** → **Anti-Cryptor**).

Anti-Cryptor component settings

Setting	Description
Scan exclusions	A *scan exclusion* is a set of conditions that must be fulfilled so that Kaspersky Embedded Systems Security will not scan a particular object for viruses and other threatening applications.
Action on threat detection	**Inform**. If this option is selected, when Kaspersky Embedded Systems Security detects a malicious encryption attempt, it logs an encryption attempt event in the event log. **Block connection for (min)**. If this option is selected, when Kaspersky Embedded Systems Security detects a malicious encryption attempt, it blocks network traffic from the device that is carrying out the malicious activity and logs an encryption attempt event in the event log. You can specify the untrusted host blocking duration in minutes. If a compromised host is blocked and you change this setting value, the blocking time for this host will not change. The blocking time is not a dynamic value, and it is calculated at the moment of blocking.
Protection scope	A *protection scope* is a list of paths to directories in which Kaspersky Embedded Systems Security monitors file activity. Kaspersky Embedded Systems Security supports environment variables and * and ? characters in masks. You can select a protection scope: **All shared network folders on the protected device**. The application monitors file activity in all directories of the server file system accessible via the Samba and NFS protocols. **Only specified shared folders**. The application monitors encryption attempts only in the specified directories.

Configuring Anti-Cryptor in the Administration Console

In the Administration Console, you can manage Anti-Cryptor settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Advanced Threat Protection** → **Anti-Cryptor**).

Anti-Cryptor component settings

Setting	Description
Enable Anti-Cryptor	This check box enables or disables the protection of files in local directories with network access by SMB/NFS protocols from remote malicious encryption. This check box is cleared by default.
Protection scopes	This group of settings contains buttons that open windows where you can configure the scan scopes and protection settings.
Exclusions	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from scans.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from scans by name mask.

Scan scopes window

The table contains the scan scopes. The application will scan files and directories located in the paths specified in the table. By default, the table contains one scan scope that includes all directories of the local file system.

Scan scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Scope name	Scan scope name.
Path	Path to the directory that the application scans.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add or configure protection scope for the Anti-Cryptor component.

Protection scope settings

Setting	Description
Scope name	Field for entering the protection scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application processes this protection scope during the component operation. If this check box is cleared, the application does not process this protection scope during the component operation. You can later include this scope in the component operation settings by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	The settings block lets you set the scan scope. You can select the file system type in the drop-down list of file systems: • **Local**, for local directories. • **Shared** displays server file system resources accessible via the Samba or NFS protocol. • **All shared** (default value) displays all server file system resources accessible via the Samba and NFS protocols.
	If **Shared** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want to add to the protection scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank.
Masks	This list contains name masks of the objects that the application scans during operation of the Anti-Cryptor component. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Protection settings window

Protection settings

Setting	Description
Action on encryption detection	The action to be performed by Kaspersky Embedded Systems Security upon detecting malicious encryption: • **Inform** user. Kaspersky Embedded Systems Security does not block the device performing encryption; it only records in the event log an event about the detection of malicious encryption. • **Block** the device performing encryption (default value).
Block untrusted host for (min)	In this field you can specify the untrusted host blocking duration in minutes. After the specified time, Kaspersky Embedded Systems Security removes the untrusted devices from the list of blocked devices. The access of the host to network file resources is restored automatically, after it is deleted from the list of untrusted hosts. If a compromised host is blocked and you change this setting value, the blocking time for this host will not change. The blocking time is not a dynamic value, and it is calculated at the moment of blocking. Possible values: integers from 1 to 2147483647. Default value: 30.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

<New exclusion scope> window

In this window, you can add and configure scan exclusion scopes.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables exclusion of the scope from scan when the application is running. If this check box is selected, the application excludes this area during scans. If this check box is cleared, the application includes this area in the scan scope. You can later exclude this scope by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	The settings block lets you set the exclusion scope. In the drop-down list of file systems, you can select the type of file system of the directories to be excluded from scans: • **Local**, for local directories. • **Mounted** – mounted directories. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
	If **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom** – resources of the device's file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a directory that you want add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Filesystem name	The field for entering the name of the file system where the directories that you want to add to the exclusion scope are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.
Masks	The list contains name masks of the objects that the application excludes from scan. Masks are only applied to objects in the directory specified in the path field. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Configuring Anti-Cryptor on the command line

In the command line, you can manage Anti-Cryptor using the Anti-Cryptor task (*Anti_Cryptor*).

By default, the Anti-Cryptor task does not run. You can start and stop this task manually.

You can configure Anti-Cryptor settings by editing the settings of the Anti-Cryptor predefined task.

Anti-Cryptor task settings

Setting	Description	Values
`ActionOnDetect`	Enables untrusted hosts blocking.	`Block` (default value) – enable untrusted hosts blocking. `Notify`: disable untrusted hosts blocking.
`BlockTime`	The time in minutes for which an untrusted device is blocked. If a compromised host is blocked, and you change a value for the `BlockTime` setting, the blocking time for this host will not change. The blocking time is not a dynamic value, and is calculated at the moment of blocking.	Integer from 1 to 4294967295. Default value: 30.
`UseExcludeMasks`	Enables exclusion of the objects specified by the `ExcludeMasks.item_#` setting from the protection scope. This setting applies only if the `ExcludeMasks.item_#` setting is specified.	`Yes` — Exclude the objects specified by the `ExcludeMasks.item_#` setting from the protection scope. `No` (default value) — Do not exclude the objects specified by the `ExcludeMasks.item_#` setting from the protection scope.
`ExcludeMasks.item_#`	Excludes objects from the protection scope by names or masks. You can use this setting to exclude an individual file from the specified protection scope by name or exclude multiple files at the same time using masks in the shell format. Before specifying a value for this setting, make sure that the `UseExcludeMasks` setting is enabled. If you want to specify several masks, specify each mask on a new line with a new index.	The default value is not defined.
The **[ScanScope.item_#]** section contains the scopes protected by the application. For the Anti-Cryptor task, you need to specify at least one protection scope; you can only specify shared directories. You can specify several [ScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of protection scope; contains additional information about the protection scope.	Default value: `All shared directories`.
`UseScanArea`	Enables protection of the specified scope. To run the task, enable protection of at least one scope.	`Yes` (default value) — Protect the specified scope. `No` — Do not protect the specified scope.
`AreaMask.item_#`	Protection scope limitation. In the protection scope, the application protects only the objects that are specified using the masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (protect all objects)

Setting	Description	Values
`Path`	Path to the directory with the objects to be protected.	`< path to local directory >` – Protect a local directory accessible via SMB/NFS. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. `AllShared` (default value) — Protect all resources accessible via SMB/NFS. `Shared:SMB` — Protect resources accessible via SMB. `Shared:NFS` — Protect resources accessible via NFS.
The **[ExcludedFromScanScope.item_#]** section contains the objects to be excluded from all [ScanScope.item_#] sections. The objects that match the rules of any [ExcludedFromScanScope.item_#] section are not scanned. The format of the [ExcludedFromScanScope.item_#] section is similar to the format of the [ScanScope.item_#] section. You can specify several [ExcludedFromScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ExcludedFromScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of the protection exclusion scope, which contains additional information about the exclusion scope.	Default value: `All objects`.
`UseScanArea`	Excludes the specified scope from protection.	`Yes` (default value) — Exclude the specified scope from protection. `No` — Do not exclude the specified scope from protection.
`AreaMask.item_#`	Limitation of the protection exclusion scope. In the exclusion scope, the application excludes only the objects that are specified using masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (exclude all objects).

Setting	Description	Values
`Path`	Path to the directory with objects excluded from protection.	`< path to local directory >` — Exclude objects in the specified directory from protection. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. `Mounted:NFS` – Exclude the remote directories mounted on a client device using the NFS protocol from protection. `Mounted:SMB` – Exclude the remote directories mounted on a client device using the Samba protocol from protection. `AllRemoteMounted` – Exclude all remote directories mounted on a client device using the Samba and NFS protocols from protection.

Managing blocked devices

While protecting a device against network threats and remote malicious encryption, Kaspersky Embedded Systems Security can block remote devices whose actions are considered to be malicious:

- If malicious encryption is detected, the application blocks access of the remote device to the shared network directories of the protected device.
- Upon detecting network attack attempts on the protected device, the application blocks network traffic from the attacking device.

You can change the blocking duration in the Network Threat Protection and Protection Against Remote Malicious Encryption settings. Once the specified period of time has elapsed, the application unblocks the device.

If you are managing the application using the command line, you can use the commands for managing blocked devices to view a list of devices that are blocked as a result of the application running on the device and manually unblock these devices before the blocking time expires. Kaspersky Security Center does not provide tools for monitoring and managing blocked devices, except for the *Network attack detected* and *Encryption detected* events.

To view the list of blocked devices, execute the following command:

```
kess-control --get-blocked-hosts
```

As a result of the command execution, the application outputs the list of blocked devices to the console.

To unblock devices, execute the following command:

```
kess-control --allow-hosts <address>
```

where `<address>` is one or more IP addresses of the devices or subnets (IPv4/IPv6, including addresses in short form). You can specify multiple IP addresses of devices or subnets by separating them with a space.

As a result of the command execution, the application unblocks the specified devices.

Examples:
IPv4 addresses:

```
dec - 192.168.0.1
dec - 192.168.0.0/24
```

IPv6 addresses:

```
hex - FEDC:BA98:7654:3210:FEDC:BA98:7654:3210
hex - FEDC:BA98:7654:3210:FEDC:BA98:7654:3210%1
hex - 2001:db8::ae21:ad12
hex - ::ffff:255.255.255.254
hex - ::
```

Application Control

The Application Control component allows you to manage the launch of applications on protected devices. Application Control lowers the risk of device infection by limiting users' access to applications.

Application launching is regulated by *Application Control rules*.

The Application Control component can operate in one of two modes:

- *Denylist*. In this mode Kaspersky Embedded Systems Security allows all users to launch any applications that are not specified in the Application Control rules. By default, the Application Control component operates in this mode.

- *Allowlist*. In this mode, Kaspersky Embedded Systems Security prevents all users from launching any applications that are not specified in the Application Control rules or signed with certificates trusted by Application Control.

Thus, if the Application Control rules are created to the fullest extent possible, Kaspersky Embedded Systems Security prohibits the launching of all new applications that are not verified by the administrator of the organization's local network, but ensures the performance of the operating system and verified applications that users need to perform their job duties.

The Kaspersky Security Center administrator or a local user with the admin role assigned in the application can allow or deny process start under the root account using the Application Control.

Application Control is disabled by default. You can enable or disable Application Control, and also configure the component's operation settings:

- Select the Application Control mode: *allowlist* or *denylist*.

- Create Application Control rules for each of the modes.

- Select the action that Kaspersky Embedded Systems Security performs upon detecting an attempt to run an application that matches the rules. Kaspersky Embedded Systems Security can *apply rules* and perform the actions specified in the rules, or *test the rules* and only inform about an attempt to run an application that matches the rules.

- Enable or disable the use of trusted certificates by Application Control and configure the list of trusted certificates.

  When using your own certificate, if it becomes compromised, you need to take steps to re-sign the protected files and replace the certificate in the policy.

  Application Control in the *allowlist mode* does not block the running of applications and scripts started using supported interpreters as long as such applications or scripts are signed with trusted certificates. The Application Control task does not control the launching of scripts from interpreters that are not supported by Kaspersky Embedded Systems Security, or the launching of scripts that are not passed to the interpreter via the command line. Kaspersky Embedded Systems Security supports the following interpreters: Java; Perl (pl); Python (py) and sh scripts (ash, bash, csh, dash, ksh, sh, tcsh, zsh).

If the interpreter is allowed to launch by the Application Control rules, Kaspersky Embedded Systems Security does not block the script launched from this interpreter. If the launch of at least one script specified in the interpreter command line is prohibited by the Application Control rules, Kaspersky Embedded Systems Security blocks all the scripts specified in the interpreter command line. Exclusion: cat script.py | python.

- Exporting and importing Application Control settings.

You can receive information about applications installed on protected devices using the Inventory task.

Application Control does not control the launch of Snap, Flatpak, or AppImage applications.

About Application Control rules

An *Application Control rule* is a set of settings that contain the conditions for triggering a rule and the actions of the Application Control component when a rule is triggered (allow or block users from starting an application):

- The application belonging to the application category. An *application category* is a group of applications that have something in common. For example, a category that includes executable files of installed applications, or a category of applications required for operation, which includes a standard set of applications used by the organization. Each category can only be used in one rule.

  Kaspersky Embedded Systems Security does not support use of the KL categories of Kaspersky Security Center.

- Allow selected users and/or user groups to run applications or prevent them from running applications. You can specify a user and/or user group that is allowed or not allowed to run applications of the specified category.

- Rule-triggering condition. A condition is represented by the following correspondence: "condition type – condition criterion – condition value". Based on the rule triggering condition, Kaspersky Embedded Systems Security applies or does not apply the rule to the application. The rules use inclusive and exclusive conditions:

  - *Inclusive conditions*. Kaspersky Embedded Systems Security applies the rule to the application if the application meets at least one inclusive condition.

  - *Exclusive conditions*. Kaspersky Embedded Systems Security does not apply the rule to the application if the application meets at least one exclusive condition or does not meet any of the inclusive conditions.

  Rule triggering conditions are created using the following criteria:

  - Name of the application's executable file.

  - Name of the directory with the application's executable file.

  - Hash of the application's executable file. Only SHA256 can be used.

  For each criterion used in the condition, a value must be specified.

You can use **masks** to specify the names of files and directories.

You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask.

You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`.

You can put a single ? character to represent any one character (including `/`) in the file or directory name.

If the settings of the application being launched match the criteria specified in the inclusive condition, the rule is triggered. In this case, Kaspersky Embedded Systems Security performs the action specified in the rule. If application settings match the criteria specified in the exclusive condition, Kaspersky Embedded Systems Security does not control the application launch.

Application control rules can have one of the following *operation statuses*:

- *Enabled*: the rule is enabled, Kaspersky Embedded Systems Security applies this rule to Application Control.
- *Disabled*: the rule is disabled and is not applied to Application Control.
- *Test* – Kaspersky Embedded Systems Security allows launching applications that meet the rule criteria, but logs information about launches of these applications in the report.

The priority of the rule operation status is higher than the priority of the action specified in the rule.

Configuring Application Control in the Web Console

In the Web Console, you can configure Application Control settings in the policy properties (**Application settings** → Security Controls → App Control).

Application Control component settings

Setting	Description
Operating mode	**Allow all except the rules list**. If this mode is selected, Application Control allows any user to run any application, except for applications that are prohibited by Application Control rules. **Block all except the rules list**. If this mode is selected, Application Control blocks any user from running any application, except for applications that are allowed by Application Control rules or signed by trusted certificates of Application Control.
Trust applications that have a digital signature	This check box enables trusted certificates for Application Control. This check box is available if the **Operating mode** is set to **Block all except the rules list**. If this check box is selected, Kaspersky Embedded Systems Security does not block applications and scripts started through supported interpreters signed with trusted certificates.
Trusted certificates	The link shows how many trusted Application Control certificates have been added. This link is available if the **Operating mode** is set to **Block all except the rules list**. Clicking the link opens a window in which you can configure the list of trusted certificates.
Action	**Apply rules**. Kaspersky Embedded Systems Security controls the launch of applications in accordance with the selected mode. **Inform**. Kaspersky Embedded Systems Security informs about an attempt to start an application.
Add policy rules to local rules	If this toggle switch is enabled, the application applies the rules specified in the policy together with the local rules configured on the protected device. If this toggle switch is disabled, the application applies only the rules specified in the policy.
Table of rules	Application Control rules used by Kaspersky Embedded Systems Security to control the launch of applications on protected devices.

Configuring Application Control in the Administration Console

In the Administration Console, you can manage Device Control settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls → Application Control**).

Application Control component settings

Setting	Description
Enable Application Control	The check box enables the Application Control component. This check box is cleared by default.
Action on application startup attempt	The action that Kaspersky Embedded Systems Security performs upon detecting an attempt to start an application that matches the configured rules: • **Apply rules** (default value). If you select this option, Kaspersky Embedded Systems Security applies Application Control rules and performs the action specified in the rules. • **Test rules**. If you select this option, Kaspersky Embedded Systems Security tests the rules and generates an event about an attempt to start an application that matches the rules.
Application Control mode	Application Control task operation mode: • **Allowlist**. If you select this option, Kaspersky Embedded Systems Security prevents all users from launching any applications that are not specified in the Application Control rules or signed with certificates trusted by Application Control. • **Denylist** (default value). If you select this option, Kaspersky Embedded Systems Security allows all users to launch any applications except those specified in the Application Control rules.
Trust applications signed by a trusted certificate	This check box enables or disables the use of the trusted certificate list by Application Control. When the check box is selected, Application Control in *allowlist* mode does not block applications that are signed with trusted certificates. This check box is available if the **Application Control** setting is set to **Allowlist**. The **Configure** button opens a window in which you can configure the list of trusted certificates for Application Control.
Application Control rules	This group of settings contains the **Configure** button. Clicking this button opens the **Application Control rules** window.
Applying rules	In the drop-down list, you can select how rules are added: • **Replace local rules with policy rules**. When you select this item, the application applies only the rules specified in the policy. • **Add policy rules to local rules** (default value). If this item is selected, the application applies rules configured in the policy together with local rules configured on the protected device.

Application Control rules window

The **Application Control rules** table contains the rules used by the Application Control component. The Application Control rules table is empty by default.

Application Control rules settings

Setting	Description
Category name	The name of the application category that is used by the rule.
Status	Operation status of the Application Control rule: • *Enabled* – the rule is enabled, Application Control applies this rule during operation. • *Disabled* – the rule is disabled and is not used when the Application Control is running. • *Test* – Application Control allows launching applications that meet the rule criteria, but logs information about launches of these applications in the report. You can change the rule status in the **Add new rule** window.

You can add, modify, or delete Application Control rules.

Adding rule window

In this window, you can configure the settings for the Application Control rule.

Adding the Application Control rule

Setting	Description
Description	Description of the Application Control rule.
Rule status	In the drop-down list, you can select the status of the Application Control rule: • *Enabled* – the rule is enabled, Application Control applies this rule during operation. • *Disabled* – the rule is disabled and is not used when the Application Control is running. • *Test* – Application Control allows launching applications that meet the rule criteria, but logs information about launches of these applications in the report.
Category	The group of settings contains the **Configure** button. Clicking this button opens the **Application categories** window.
Users and their rights	The table contains a list of users or user groups to which the Application Control rule applies, and the types of access assigned to them, and consists of the following columns: • **User or group name** – names of users or names of user groups to which the Application Control rule applies. • **Access** – the type of access: **Allow** launching the applications or **Block** launching the applications. You can add, edit, or delete users or user groups.

Application categories window

In this window, you can add a new category or configure the category settings for an Application Control rule.

Kaspersky Embedded Systems Security does not support use of the KL categories of Kaspersky Security Center.

Application Control categories

Setting	Description
Category name	List of the added Application Control categories.
Add	Clicking the button starts the category creation wizard. Follow the instructions of the Wizard. For details about creating a category, refer to the Kaspersky Security Center Help.
Edit	Clicking this button opens the category properties window, where you can change the category settings. The **Golden Image (local)** category cannot be edited.

User or group window

In this window, you can specify a local or domain user or user group for which you want to configure a rule.

Setting	Description
Type	The **User** or **Group** to which the Application Control rule applies.
User or group name	Name of the user or user group to which the Application Control rule applies.
Access	Access type: **Allow** launching the applications or **Block** launching the applications.

Trusted certificates of Application Control window

You can configure a list of certificates that will be trusted by Application Control. Application Control allows running applications signed by certificates from this list.

The following information is displayed for each certificate:

- Certificate subject
- Certificate serial number
- Certificate issuer
- Certificate start date
- Certificate expiration date
- SHA256 certificate fingerprint

By default, the certificate list is empty.

You can add or remove certificates.

Clicking the **Add** button opens a window in which you can add a certificate to the list of trusted certificates in one of the following ways:

- Indicate the path to the certificate file. The **Browse** button opens the standard file selection window. Indicate the path to the file that contains the certificate, in DER or PEM format.
- Copy the contents of the certificate file to the **Enter certificate details** field.

Configuring Application Control on the command line

In the command line, you can manage Application Control by using the Application Control predefined task (*Application_Control*).

By default, the Application Control task does not run. You can start and stop the task manually.

You can configure Application Control on a device by editing the settings of the Application Control predefined task.

If you change the list of allowed applications or prohibit the launch of all applications or applications affecting Kaspersky Embedded Systems Security's operation, then when modifying the task settings using the configuration file or using command line options, run the `kess-control --set-settings` command with the `--accept` flag.

You can also configure Application Control using Application Control commands:

- Create and edit lists of categories.
- View the list of categories created in the application.
- Configure the list of application control rules.
- Configure the list of certificates trusted by Application Control.

Application Control task settings

The table describes all possible values and the default values of all the settings that you can specify for the Application Control task.

Application Control task settings

Setting	Description	Values
`AppControlMode`	Application Control task operation mode.	`AllowList` – In this mode, Kaspersky Embedded Systems Security prevents all users from launching any applications that are not specified in the Application Control rules or signed with certificates trusted by Application Control. `DenyList` (default value) – Kaspersky Embedded Systems Security allows users to launch any applications that are not specified in the Application Control rules.
`UseTrustedCustomCerts`	Enable trusted certificates for Application Control.	`Yes` (default value) – make Application Control use trusted certificates. Application Control in *allowlist* mode does not block applications that are signed with trusted certificates. You can configure the list of trusted certificates for Application Control using application commands. `No` – make Application Control ignore the list of trusted certificates.
`AppControlRulesAction`	The action that Kaspersky Embedded Systems Security performs upon detecting an attempt to start an application that matches the configured rules.	`ApplyRules` (default value) – Kaspersky Embedded Systems Security applies Application Control rules and performs the action specified in the rules. `TestRules` – Kaspersky Embedded Systems Security tests the rules and generates an event about the detection of an application that matches the rule.
The **[Categories.item_#]** section contains the following settings:		
`Name`	Name of the application category to which the rule applies.	
`UseIncludes`	Usage of inclusive conditions to trigger the rule.	`Yes` – apply the rule to the application if the application meets at least one inclusive condition. `No` (default value) – do not apply the rule to the application, even if the application meets the inclusive conditions.

Setting	Description	Values
`IncludeFileNames.item_#`	Name of the executable file that triggers the rule.	You can use **masks** to specify the file name. You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask. You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`. You can put a single ? character to represent any one character (including `/`) in the file or directory name.
`IncludeFolders.item_#`	Name of the directory with the application's executable file that triggers the rule.	You can use **masks** to specify the directory name. You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask. You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`. You can put a single ? character to represent any one character (including `/`) in the file or directory name.
`IncludeHashes.item_#`	SHA256 hash of the executable file that triggers the rule.	Only SHA256 can be used.
`UseExcludes`	Usage of excluding conditions to trigger the rule.	`Yes` – do not apply the rule to the application if the application meets at least one exclusive condition or does not meet any of the inclusive conditions. `No` (default value) – apply the rule to the application, even if the application meets at least one exclusive condition.
`ExcludeFileNames.item_#`	Name of the executable file that triggers the rule.	You can use **masks** to specify the file name. You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask. You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`. You can put a single ? character to represent any one character (including `/`) in the file or directory name.
`ExcludeFolders.item_#`	Name of the directory with the application's executable file that triggers the rule.	You can use **masks** to specify the directory name. You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask. You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`. You can put a single ? character to represent any one character (including `/`) in the file or directory name.
`ExcludeHashes.item_#`	SHA256 hash of the executable file that triggers the rule.	Only SHA256 can be used.
The **[AllowListRules.item_#]** section contains a list of Application Control rules for the *AllowList* operation mode. Each [AllowListRules.item_#] section contains the following settings:		
`Description`	Description of the Application Control rule.	
`AppControlRuleStatus`	Operation status of the Application Control rule:	`On` (default value): the rule is enabled, Kaspersky Embedded Systems Security applies this rule to Application Control. `Off`: the rule is not used for the Application Control. `Test` – Kaspersky Embedded Systems Security allows applications covered by the rule to be launched, but logs information about the launch of these applications in the report.

Setting	Description	Values
`Category`	Name of the application category for which the rule applies. You can specify the "Golden Image" category.	
The **[AllowListRules.item_#.ACL.item_#]** section contains a list of users who are allowed or denied to run applications.		
`Access`	Access type assigned to a user or user group.	`Allow` (default value) — Allow running applications. `Block` – Deny running applications.
`Principal`	User or user group to which the Application Control rule applies.	`\Everyone` (default value): the rule applies to all users. `< user name >` is the name of the user to which the rule applies. `@< group name >` is the name of the group of users to which the rule applies.
The **[DenyListRules.item_#]** section contains a list of Application Control rules for the *DenyList* operation mode. Each [DenyListRules.item_#] section contains the following settings:		
`Description`	Description of the Application Control rule.	
`AppControlRuleStatus`	Operation status of the Application Control rule:	`On` (default value): the rule is enabled, Kaspersky Embedded Systems Security applies this rule to Application Control. `Off`: the rule is not used for the Application Control. `Test` – Kaspersky Embedded Systems Security allows applications covered by the rule to be launched, but logs information about the launch of these applications in the report.
`Category`	Name of the created application category to which the rule applies. You can specify the "Golden Image" list of applications as a category.	
The **[DenyListRules.item_#.ACL.item_#]** section contains a list of users who are allowed or denied to run applications.		
`Access`	Access type assigned to a user or user group.	`Allow` – allow applications to start. `Block` (default value) – do not allow applications to start.
`Principal`	User or user group to which the Application Control rule applies.	`\Everyone` (default value): the rule applies to all users. `< user name >` is the name of the user to which the rule applies. `@< group name >` is the name of the group of users to which the rule applies.

Creating and editing a list of categories

You can create a new category in two ways:

- Using the "kess --set-settings" command and the Application Control task settings configuration file (Application_Control)
- Using the "kess --set-categories" command and the category settings configuration file

To create application categories, run the following command:

```
kess-control --set-categories --file < path to configuration file >
```

where:

`--file < path to configuration file >` is the path to the configuration file with the category settings.

The file with category settings must have the following structure:

```
[
    {
         "Exclude" : [ "(FilePath like <full path to the executable file of the
   application>)", "(FileHash == <hash of the executable file>)" ],
         "GUID" : "<unique category ID>",
         "Include" : [ "(FilePath like <full path to the executable file of the
   application>)", "(FileHash == <hash of the executable file>)" ],
         "Name" : "<name of category 1>"
     },
   {
         "Exclude" : [ "(FilePath like <full path to the executable file of the
   application>)", "(FileHash == <hash of the executable file>)" ],
         "GUID" : "<unique category ID>",
         "Include" : [ "(FilePath like <full path to the executable file of the
   application>)", "(FileHash == <hash of the executable file>)" ],
         "Name" : "<name of category 2>"
     }
]
```

To specify the file name in the `Exclude` and `Include` fields, you can use **masks**.

You can use the * character (any sequence of characters) or the ? character (any one character) as the file or directory name mask.

You can put the * character to represent any set of characters (including an empty set) in a file or directory name that includes the / character. For example, `/dir/*/file*/` or `/dir/file*/`.

You can put a single ? character to represent any one character (including `/`) in the file or directory name.

> The `Name` setting is required. If you do not specify the name of the category, it will not be created or will be deleted. The `GUID` setting is also required. If you do not specify it, an error message is displayed and the category is not created. The `GUID` setting must be specified without hyphens.

To edit the list of created application categories, run the following command:

```
kess-control --set-categories [--names <name of category 1> <name of category 2> …
<name of category N>] --file <path to configuration file>
```

where:

- `<name of category 1> <name of category 2> … <name of category N>` are names of categories for which you want to change information. If you want to change information about several categories, specify the names of the categories, separated by a space. If you do not specify a category name, existing categories are deleted and new categories are created from the specified file.
- `--file <path to configuration file>` is the path to the configuration file with the category settings.

Viewing the list of categories

On the command line, you can view the list of created application categories using the Application Control command, `kess-control --get-categories`.

The list of created categories contains the following categories:

- Categories created in Kaspersky Security Center.
- Categories added in the Application Control task settings using the command line.
- The "GoldenImage" category created using the Inventory task (in Kaspersky Security Center or using the command line).

To view the list of all created application categories, run the following command:

```
kess-control --get-categories [--file <path to configuration file>] [--json]
```

where:

- `--file <path to configuration file>` – full path to the JSON configuration file to which the settings will be output.
- `--json` is specified to output the settings in JSON format. If the --json option is omitted, the settings are output in the INI format.

Kaspersky Embedded Systems Security displays the following information about each application category:

- Unique identifier (GUID) of the category
- Category name
- list of inclusive conditions to trigger the rule
- list of exclusive conditions to trigger the rule

To view the list of separate application categories, execute the following command:

```
kess-control --get-categories [--names <category name 1> <category name 2> ...
<category name N>] [--file <path to configuration file>] [--json]
```

where:

- `<category name 1> <category name 2> ... <category name N>` are the names of the categories for which you want to view information. If you want to view information about several categories, specify the names of the categories, separated by a space.
- `--file <path to configuration file>` – full path to the JSON configuration file to which the category list will be exported.
- `--json` is specified to output the settings in JSON format. If the --json option is omitted, the settings are output in the INI format.

If in the Application Control task settings, in the `[Categories.item_#]` section for inclusive or exclusive conditions for triggering a rule, you specify symbolic links to an application file or directory with executable files, then when viewing the list of categories for these conditions, the source path to which the symbolic link points is displayed.

Configuring the Application Control rule list

To view the list of Application Control rules, run the following command:

```
kess-control --get-settings 21 [--file <path to configuration file>] [--json]
```

where:

- `--file <path to configuration file>` is the full path to the configuration file to which you want to output the settings.
- `--json`: output data in JSON format.

Kaspersky Embedded Systems Security displays the following information about Application Control rules:

- Application Control task operation mode.
- Action that Application Control takes upon detecting an attempt to launch an application that matches the configured rule.
- Description of the Application Control rule (if any).
- Operation status of the Application Control rule.
- Name of the application category the rule applies to.
- Access type assigned to a user or user group.
- Uuser or user group to which the Application Control rule applies.

To edit the list of application categories and Application Control rules, run the following command:

```
kess-control --set-settings 21 [--file <path to configuration file>] [--json]
```

where:

- `--file <path to configuration file>` is the full path to the configuration file from which you want to import the settings.
- `--json` – import data from a JSON file.

To delete the list of application categories and Application Control rules, run the following command:

```
kess-control --set-settings 21 --set-to-default
```

Managing the list of trusted certificates of Application Control

To add a certificate to the trusted certificate list for Application Control, run the following command:

```
kess-control --add-app-control-trust-certificate <path to certificate>
```

where:

`<path to certificate>` is the path to the certificate file that you want to add (PEM or DER format).

To remove a certificate from Application Control's trusted certificate list, run the following command:

```
kess-control --remove-app-control-trust-certificate <serial number of certificate>
```

To view Application Control's list of trusted certificates, run the following command:

```
kess-control --query-app-control-trust-certificates
```

The following information is displayed for each certificate:

- Certificate subject
- serial number
- Certificate issuer
- Certificate start date
- Certificate expiration date
- SHA256 certificate fingerprint

When checking the application signature using a trusted certificate, the validity period of this certificate is not checked.

Exporting and importing Application Control settings

To simplify the configuration of the Application Control component on user devices, you can export and import the Application Control settings and the list of application categories.

Exporting and importing Application Control settings in the Web Console

In the Web Console, you can export or import Application Control settings and the list of application categories in the properties of a managed device (**Assets (Devices)** → **Managed devices** → <device name> link → **Applications** tab → <name of the Kaspersky Embedded Systems Security application> link → **Application settings** tab → **Application Control** section).

Clicking the **Export** button under **Application Control settings** opens a window in which you can specify the file to which you want to save the settings. Clicking the **Import** button under **Application Control settings** opens a window in which you can specify the file from which you want to import the settings.

Clicking the **Export** button under **Application categories** opens a window in which you can specify the file to which you want to save the added application categories. Clicking the **Import** button under **Application categories** opens a window in which you can specify the file from which you want to import application categories.

Exporting and importing Application Control settings in the Administration Console

In the Administration Console, you can export and import Application Control settings and the list of application categories in the properties of the managed device (**Managed devices** → **Devices** tab → device properties window → **Applications** section → <application name> → **Properties** button → **<application name> settings** window → **Application Control** section).

Clicking the **Export** button under **Application Control settings** opens a window in which you can specify the file to which you want to save the settings. Clicking the **Import** button under **Application Control settings** opens a window in which you can specify the file from which you want to import the settings.

Clicking the **Export** button under **Application categories** opens a window in which you can specify the file to which you want to save the added application categories. Clicking the **Import** button under **Application categories** opens a window in which you can specify the file from which you want to import application categories.

Exporting and importing Application Control settings on the command line

On the command line, you can export or import Application Control settings and the list of application categories using Application Control management commands.

Inventory

The Inventory task provides information about all applications executable files stored on the client devices. Obtaining information about the applications installed on the devices can be useful, for example, for creating Application Control rules.

You can configure the following inventory settings:

- Select the types of objects that the application will detect on the device during inventory (files, scripts).
- Configure inventory scopes (paths to directories in which to search for executable application files).
- Configure exclusions from the inventory.
- Select the action that Kaspersky Embedded Systems Security must perform with the "Golden Image" application category upon completion of the Inventory task. You can add applications that the task discovers on the device to the category, leave the category unchanged, or delete it.
- Enable or disable the transmission of information about applications that the Inventory task discovers on the device to the Kaspersky Security Center Administration Server. If information about applications on the device is transmitted to the Administration Server, it can be used to configure application categories for Application Control rules using the Kaspersky Security Center.

Configuring Inventory in the Web Console

In the Web Console, you can perform an inventory of the applications for the protected device using the *Inventory* task.

You can create and run Inventory user tasks. You can configure inventory settings by editing the settings of these tasks.

The Kaspersky Security Center database can store information about up to 150,000 processed files. When this number of records is reached, new files will not be processed. To resume the Inventory task, delete the files registered in the Kaspersky Security Center database as a result of previous inventories, from the device where Kaspersky Embedded Systems Security is installed.

Inventory task settings

Setting	Description
Action to perform on the "Golden image" category upon task completion	In the drop-down list, you can select the action that Kaspersky Embedded Systems Security must perform with the "Golden Image" application category upon completion of the Inventory task: • **No changes** – do not change the "Golden Image" application category after completing the inventory. This is the default. • **Update or create (if does not exist)** – add applications that the Inventory task discovers on the device to the "Golden Image" category. If the category does not exist, it is created. If the category exists, the list of applications in the category is updated. • **Delete** – delete the "Golden Image" category after completing the inventory. You can use the "Golden Image" category in Application Control rules.
Scan all executables	This check box enables or disables of executable file scans. This check box is selected by default.
Scan binaries	The check box enables or disables scans of binary files (files with the extensions .elf, .java and .pyc). This check box is selected by default.

Setting	Description
Scan scripts	This check box enables or disables script scans. This check box is selected by default.
Inventory scopes	The table contains the inventory scopes scanned by the application. The application will scan files and directories located in the paths specified in the table. By default, the table contains one inventory scope – /usr/bin. You can add, edit, delete, move up, or move down inventory scopes in the table. You can also import the list of inventory scopes from a file by clicking **Import** or export the list of inventory scopes to a file in JSON format by clicking **Export**. When importing, you are prompted to replace the list of inventory scopes or add inventory scopes to the existing list.

Add scan scope window

In this window, you can add and configure scan scope for the Inventory task.

Inventory scope settings

Setting	Description
Scope name	Field for entering the inventory scope name. This name will be displayed in the table in the **Scan settings** section. The entry field must not be blank.
Use this scope	This check box enables or disables the scan of this scope when the task is performed. If this check box is selected, the application processes this inventory scope while running the task. If this check box is cleared, the application does not process this inventory scope while running the task. You can later include this scope in task settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to include in the inventory scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank. The **/** path is specified by default – the application scans all directories of the local file system.
Masks	This list contains name masks of the objects that the application scans while running the task. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusion scopes section

In the **Exclusion scopes** section for the Inventory task, you can configure the scopes to be excluded from scans.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

Add exclusion scope window

In this window, you can add and configure scan exclusion scope for the Inventory task.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables the exclusion of the scope when the task is executed. If this check box is selected, the application excludes this scope during task execution. If this check box is cleared, the application includes this scope during task execution. You can later exclude this scope from scanning by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to exclude from the inventory. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank.
Masks	The list contains name masks of the objects that the application excludes from scan. You can add, edit, or delete masks.

Configuring Inventory in the Administration Console

In the Kaspersky Security Center Administration Console, you can perform an inventory of the applications for the protected device using the *Inventory* task.

You can create and run Inventory user tasks. You can configure the scan settings by editing the settings of the tasks.

The Kaspersky Security Center database can store information about up to 150,000 processed files. When this number of records is reached, new files will not be processed. To resume the Inventory task, delete the files registered in the Kaspersky Security Center database as a result of previous inventories, from the device where Kaspersky Embedded Systems Security is installed.

Inventory task settings

Setting	Description
Action to perform on the "Golden image" category upon task completion	In the drop-down list, you can select the action that Kaspersky Embedded Systems Security must perform with the "Golden Image" application category upon completion of the Inventory task: • **No changes** – do not change the "Golden Image" application category after completing the inventory. This is the default. • **Update or create (if does not exist)** – add applications that the Inventory task discovers on the device to the "Golden Image" category. If the category does not exist, it is created. If the category exists, the list of applications in the category is updated. • **Delete** – delete the "Golden Image" category after completing the inventory. You can use the "Golden Image" category in Application Control rules.
Scan all executables	This check box enables or disables of executable file scans. This check box is selected by default.
Scan binaries	The check box enables or disables scans of binary files (files with the extensions .elf, .java and .pyc). This check box is selected by default.
Scan scripts	This check box enables or disables script scans. This check box is selected by default.
Inventory scopes	The group of settings contains the **Configure** button. Clicking this button opens the **Scan scopes** window.

In the **Exclusions** section for the Inventory task, you can also configure scopes to be excluded from scans.

Scan scopes window

The table contains the scan scopes. The application will scan files and directories located in the paths specified in the table. By default, the table contains one scan scope – /usr/bin.

Scan scope settings for the Inventory task

Setting	Description
Scope name	Scan scope name.
Path	Path to the directory that the application scans.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure scan scope for the Inventory task.

Inventory scope settings

Setting	Description
Scan scope name	Field for entering the scan scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables the scan of this scope when the task is performed. If this check box is selected, the application processes this scan scope while running the task. If this check box is cleared, the application does not process this scan scope while running the task. You can later include this scope in task settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to include in the scan scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank.
Masks	This list contains name masks of the objects that the application scans while running the task. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusions section

Settings of scan exclusions

Group of settings	Description
Exclusion scopes	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from monitoring.

Exclusion scopes window

This window contains scan exclusion scopes. The application does not scan files and directories located at the paths specified in the table.

Exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	The status indicates whether the application uses this exclusion.

By default, the table of exclusion scopes is empty. You can add, edit, and delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**.

<New exclusion scope> window

In this window, you can add and configure scan exclusion scope for the Inventory task.

Exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables the exclusion of the scope when the task is executed. If this check box is selected, the application excludes this scope during task execution. If this check box is cleared, the application includes this scope during task execution. You can later exclude this scope from scanning by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to exclude from the inventory. You can use **masks** to specify the path. The field must not be blank. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name.
Masks	The list contains name masks of the objects that the application excludes from scan. You can add, edit, or delete masks.

Configuring Inventory on the command line

You can use the command line to inventory applications on the protected device as follows:

- With the help of the Inventory_Scan *predefined task*. You can manually start or stop this task, and configure the task run schedule. You can configure scan settings by editing the settings of this task.
- With the help of user inventory tasks (tasks of the *InventoryScan* type). You can manually start, stop, pause, or resume user tasks and configure the task schedule.

You can view the list of applications detected on the device as a result of the Inventory task by using Application Control management commands.

Inventory task settings

The table describes all possible values and the default values of all the settings that you can specify for the Inventory task.

Inventory task settings

Setting	Description	Values
`ScanScripts`	Enables script scanning.	`Yes` (default value) — Scan scripts. `No` — Do not scan scripts.
`ScanBinaries`	Enables binary files scanning (elf, java, and pyc).	`Yes` (default value) — Scan binaries. `No` — Do not scan binaries.
`ScanAllExecutable`	Enables the scanning of files with an executable bit.	`Yes` (default value) — Scan files with an executable bit. `No` — Do not scan files with an executable bit.
`GoldenImageAction`	The action that Kaspersky Embedded Systems Security must perform with the "Golden Image" application category upon completion of the Inventory task: You can use the "Golden Image" category in Application Control rules.	`DoNothing` (default) – do not change the "Golden Image" application category after completing the inventory. `Create` – add detected applications to the "Golden Image" application category. If the category does not exist, it is created. If the category exists, the list of applications in the category is updated. `Remove` – delete the "Golden Image" category after completing the inventory.
The **[ScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the inventory scope. The maximum length of the string specified using this setting is 4096 characters.	Default value: `All objects`.
`UseScanArea`	Enables scans of the specified inventory scope. To run the task, enable scans of at least one inventory scope.	`Yes` (default value) — Scan the specified inventory scope. `No` — Do not scan the specified inventory scope.
`AreaMask.item_#`	Inventory scan scope limitation. In the inventory scan scope, the application scans only the files that are specified using the masks in the shell format. If this setting is not specified, the application scans all the objects in the inventory scope. You can specify several values for this setting.	The default value is `*` (scan all objects).
`Path`	Path to the directory with objects to be scanned.	`< path to local directory >` — scan objects in the specified directory. Default value: `/usr/bin`
The **[ExcludedFromScanScope.item_#]** section contains the following settings:		
`AreaDesc`	Description of the inventory exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from the inventory.	`Yes` (default value) — Exclude the specified scope. `No` — Do not exclude the specified scope.
`AreaMask.item_#`	Limiting the inventory exclusion scope using shell masks. If this setting is not specified, the application excludes all the objects in the inventory scope. You can specify several values for this setting.	Default value: `*` (exclude all objects)

Setting	Description	Values
`Path`	Path to the directory with objects to be excluded.	`< path to local directory >` excludes objects in the specified directory from the scan. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name.

Viewing a list of detected applications

To view the list of applications detected on the device, execute the following command:

```
kess-control --get-app-list [--json]
```

where `--json` means output data in JSON format.

Kaspersky Embedded Systems Security displays the following information about the detected applications:

- **Date and time of inventory**. Date and time when the Inventory task was performed
- **Number of applications**. The number of applications detected on the device
- The list of applications containing the following information:
  - **Path**. Path to the application.
  - **Hash**. Application hash sum.
  - **Type**. Application type. For example, `Script`, `Executable`.
  - **Categories**. Categories that the application belongs to (if they were previously created). You can view the list of created application categories using the command `kess-control --get-categories`.

When you add a new category, its information is not automatically updated in the application list. To update the application list, you need to restart the Inventory task.

Device Control

The *Device Control* component allows you to manage user access to the devices that are installed on or connected to the client device (for example, hard drives, cameras, or Wi-Fi modules). Access management lets you protect the client device from infection when external devices are connected, and prevent data loss or leaks.

The Device Control component is enabled automatically with the default settings when Kaspersky Embedded Systems Security is started.

Device Control manages user access on the following levels:

- **Device type** as classified by Device Control, such as printers, removable drives, or CD/DVD drives. One of the following access modes can be applied to each device type:
  - *Allow*, to allow access to devices of this type.
  - *Block*: block access to devices of this type.
  - *Depends on bus*: allow or block access to devices depending on the access mode set for the bus through which the device is connected.
  - *By rule*: allow or block access to devices depending on the access rules. A *device access rule* is a set of options that determine which users can access devices that are installed on the client device or connected to it, and at what time. This access mode can be specified only for storage devices (hard drives, removable drives, floppy disk drives, CD/DVD drives). When a forbidden device is connected, the application denies access to the device to the users specified in the rule and displays a notification. During attempts to read and write on this device, the application silently blocks the users specified in the rule from reading/writing. If you try to perform an operation with a device whose access mode is set to *By rule*, but no rule active at the time of access is found, the operation is blocked.
  - *Read only*: allow read-only access to devices of the selected type, while blocking write access. This access mode can be specified only for storage devices (hard drives, removable drives, floppy disk drives, CD/DVD drives). If the device has the "read-only" access type, when the trusted device is assigned to a specific user, the assigned user is ignored and the device is made available for writing to all users.
- **Connection bus**. *Connection bus* is an interface that devices use to connect to the client device, such as USB or FireWire. One of the following access modes can be applied to connection buses:
  - *Allow*: grant access to devices connected through this connection bus.
  - *Block*: deny access to devices connected using this connection bus.

  For example, access may be denied to all devices connected via USB.

By default, the *Depends on connection bus* access mode is selected for all device types. The *Allow* access mode is selected for connection buses. Device Control grants users full access to all devices accordingly.

When Device Control is enabled for the first time, it generates a *DeviceAllowed* event for all detected devices with a known device or bus type. No repeat events are generated upon subsequent component runs unless there were changes in the control settings for these devices.

When Device Control is disabled, the application unblocks access to blocked devices.

Device Control does not block system drives. If the application cannot automatically detect the system drive, the Device Control component terminates with an error.

Blocking devices by device type or connection bus via the system device driver is not supported on the following Linux kernels: 3.10, 5.14, 5.15, 5.17, 6.1, 6.8. On these kernels and in the *By rule* access mode, only the opening of files and reading of directories (that is, getting the names of files and directories) are blocked. On systems that do not support fanotify, blocking the reading of directories is also not supported.

Device Control ignores mount point exclusions. Access to a device mounted at an excluded point can be limited with Device Control settings.

You can enable, disable, and configure Device Control:

- Select the application's operation mode when there is an attempt to access a device to which access is prohibited by Device Control settings: block or only notify about the attempt to access the device.
- Select a device access mode depending on the type.
- Select a device access mode depending on the bus the devices are connected to.
- Remove individual devices from the scope of Device Control by adding them to the list of trusted devices. *Trusted devices* are devices to which users have full access. For example, you can limit access to specific USB devices or only to USB drives; access to other USB devices is denied. Additionally, you can specify users or groups for which devices are to be trusted.

  You can manage the list of trusted devices on the command line or using Kaspersky Security Center.

  You can add devices to a list of trusted devices by identifier or identifier mask. You can get information about devices installed on client devices or connected to client devices using the command line on a client device or using Kaspersky Security Center. Information sharing with the Kaspersky Security Center Administration Server is enabled by default.

  > Information about devices is transferred if the client device is under the control of an active policy and synchronized with Network Agent (performed with the frequency specified in the Network Agent policy properties, by default – every 15 minutes).

  If you manage the application using Kaspersky Security Center, you can also use the following to add devices to the trusted list:

  - List of devices discovered on managed client devices. The list is displayed in the policy properties in the Web Console or the Administration Console.
  - List of devices generated based on Device Control events in Kaspersky Security Center. You can create this list in the policy properties in the Administration Console.

  In the Administration Console, you can also export the configured list of trusted devices to a file or import a list of trusted devices from a file.
- Configure an access schedule for storage devices (hard drives, removable drives, floppy disk drives, CD/DVD drives).

  > In general application settings, if blocking access to files during scans is disabled, you cannot use a device access schedule to block access to devices.

- You can define access rules for devices depending on their type. Allow or block access for specified users at a specified time.
- Allow the user to request temporary access to a blocked device and grant the user access to the device.

> Granting access to a blocked device is available only if Kaspersky Security Center 16.2 is deployed in the organization and the client device is able to establish a connection with the Administration Server.

Configuring Device Control in the Web Console

In the Web Console, you can configure Device Control settings in the policy properties (**Application settings** → **Security Controls** → **Device Control**)

Device Control settings

Setting	Description
Trusted devices	List of trusted devices and users who are granted access to these devices. Kaspersky Embedded Systems Security supports `*` and `?` characters in masks.
Message templates	**Message to user**. Template of the message that appears when a user attempts to access a blocked device. This message also appears when a user attempts to perform an operation on the device contents that was blocked for this user.
Allow requesting temporary access	If this check box is selected, the user can request temporary access to the blocked device in the graphical interface of the application or on the command line.
Operating mode	Kaspersky Embedded Systems Security perform the selected action when an attempt is made to gain access to a device to which access is denied in accordance with the Device Control settings: **Inform**. If you select this action, Kaspersky Embedded Systems Security tests the selected access mode and generates an event about detection of an attempt to access a device. **Block**. When this action is selected, Kaspersky Embedded Systems Security applies the access mode defined for the device or bus.
Devices and Wi-Fi networks	List of all possible device types according to the classification of the Device Control component and the mode of access to them.
Connection buses	List of all possible connection buses according to the classification of the Device Control component and the mode of access to them. Kaspersky Embedded Systems Security allows or denies access to devices via the connection bus if the **Depends on bus** mode is selected for the device.

Configuring Device Control in the Administration Console

In the Administration Console, you can manage Device Control settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls** → **Device Control.**).

Device Control settings

Setting	Description
Enable Device Control	This check box enables or disables Device Control. This check box is selected by default.
Trusted devices	This group of settings contains the **Configure** button. Clicking this button opens the **Trusted devices** window. In this window, you can add a device to the list of trusted devices.

Setting	Description
Device Control operating mode	Response to attempts to access a device that is restricted according to Device Control rules: • **Inform**. If you select this option, Kaspersky Embedded Systems Security tests the selected access mode and generates an event about detection of an attempt to access a device. • **Block** (default value). When this option is selected, Kaspersky Embedded Systems Security applies the access mode defined for the device or bus.
Device Control settings	This group of settings contains buttons that open windows where you can configure access mode for devices by type and connection buses.
Allow requesting temporary access	If this check box is selected, the user can request temporary access to the blocked device in the graphical interface of the application or on the command line.
Message templates	This group of settings contains the **Configure** button. Clicking this button opens the **Message templates** window.

Trusted devices window

The table contains a list of trusted devices. The table is empty by default.

Trusted device settings

Setting	Description
Device ID	ID of a trusted device.
Device name	Name of a trusted device.
Device type	Trusted device type (for example, Hard drive or Smart card reader).
Host name	Name of the client device the trusted device is connected to.
User or group name	Name of the user or group for which the device is trusted.
Comment	Comment related to a trusted device.

You can add a device to the trusted device list by clicking the **Add** button as follows:

- Specify a device ID or device ID mask (**Add** → **Add device by ID or mask**).
- Select the desired device from the list of devices detected on managed devices (**Add** → **Select a device from the list of devices detected on hosts**).
- Select the desired device from the list of devices detected in Device Control events in Kaspersky Security Center (**Add** → **Add device based on events**).

You can edit and delete trusted devices in the table.

You can also import the list of devices from a file by clicking **Advanced** -> **Import** and export the list of added devices to a file in JSON format by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**. When importing, you will be prompted to replace the list of trusted devices or add the devices to the existing list.

Three import modes are possible:

- **Combine with existing** to merge the lists of trusted devices upon import. If the file already has an entry for this device ID, settings of the existing entry are updated. For devices that were not previously on the list, entries are added.
- **Add to existing** to add trusted devices to the list upon import. Only entries for devices not previously on the list of devices are added to the list.
- **Replace existing** to replace the list of trusted devices. The existing list of trusted devices is replaced with the list from the file.

Trusted device window

In this window, you can add a device to the list of trusted devices by its identifier.

Adding device by ID

Setting	Description
Device ID	The field for entering the identifier or the identifier mask of the device that you want to add to the list of trusted devices. To specify an identifier, you can use the following wildcards: * (any sequence of characters) or ? (any single character). For example, you can specify the USBSTOR* mask to allow access to all USB drives.
Find on hosts	Clicking the button displays the devices found on the connected client devices using the specified ID or mask. The button is available if the **Device ID** field is not empty.
Devices found	The table contains the following columns: • **Device type** – type of device found (for example, Hard drive or Smart card reader). • **Device ID** – ID of the device found. • **Device name** – name of the device found. • **Host name** — name of the client device that the found device is connected to.
Comment	The field for entering a comment for the device that you want to add to the list of trusted devices (optional).
Allow access for users and/or groups of users	This field displays the name of the user and/or group for which the device is trusted. The default value is **\Everyone** (all users).
Configure	Clicking the button opens the Access to trusted device window.

Device window on client devices

In this window you can add a device to the list of trusted devices by selecting it in the list of existing managed devices.

Information about existing devices is available only if an active policy has been created and synchronization with the Network Agent has been performed (the synchronization interval is specified in the Network Agent policy, 15 minutes by default).

When you create a new inactive policy, the list will be empty.

Adding a device from the list of devices detected on managed client devices

Setting	Description
Host name	Field for entering the name or the name mask for the managed device for which you want to find connected devices. The default mask is * – all managed devices.
Device type	In this drop-down list, you can select the type of connected device to search for (for example, Hard drives or Smart card readers). The **All devices** option is selected by default.
Device ID	Field for entering the identifier or identifier mask for the device you want to find. The default mask is * – all devices.
Find on hosts	When you click this button, the application searches for devices with the specified parameters. The search results are displayed in the table below.
List of detected devices	The table contains the following columns: • **Host name** — name of the client device that the found device is connected to. • **Device type** – type of device found (for example, Hard drive or Smart card reader). • **Device name** – name of the device found. • **Device ID** – ID of the device found.

List of devices based on Device Control events window

In this window, you can create a list of devices detected in Device Control events in Kaspersky Security Center. You can then add devices from this list to the list of trusted devices.

You can select one or more devices to add to the list of trusted devices.

Information about devices is available if Device Control is enabled, there is an active policy, and synchronization with Network Agent has been performed (synchronization is performed at the frequency specified in the Network Agent policy; by default – every 15 minutes).

When you create a new inactive policy, the list will be empty.

Adding devices from the list of devices detected in Kaspersky Security Center events

Setting	Description
Event types	You can use check boxes to select the types of Device Control events sent to Kaspersky Security Center. The list includes the devices detected in events of the specified types. All check boxes are selected by default.
Administration group	A field for entering the name of the administration group containing the client devices that the detected devices are connected to. Only devices connected to client devices from the specified administration group are included in the list. You can specify part of the group name. By default, the list includes devices connected to client devices in any administration group.
Events from period	In the drop-down list, you can select the time period when the events occurred. The list includes devices that were detected in events sent to Kaspersky Security Center during the specified period. By default, the list includes devices detected in events over the last month.
Search for devices	When you click this button, the application searches for devices with the specified parameters. The search results are displayed in the table below.
Devices found	The table contains the following columns: • **Device type** – type of device found (for example, Hard drive or Smart card reader). • **Device ID** – ID of the device found. • **Device name** – name of the device found. • **Host name** — name of the client device that the found device is connected to. • **Connection date** is the date when the found device was connected. • **Administration group** is the name of the administration group containing the client device that the detected device is connected to.
Comment	The field for entering a comment for the device that you want to add to the list of trusted devices (optional).
Allow access for users and/or groups of users	This field displays the name of the user and/or group for which the device is trusted. The default value is **\Everyone** (all users).
Configure	Clicking the button opens the **Access to trusted device** window.

Access to trusted device window

In this window, you can configure access of users or user groups to a selected trusted device.

Access to trusted device

Setting	Description
Users and/or user groups	List of users and groups for which you want to make the added device trusted. By default, the table contains the **\Everyone** item (all users). You can add, edit, or delete users or user groups.

User or group window

In this window, you can specify a user or a user group for which you want to make the device trusted.

Trusted device settings

Setting	Description
Type	The **user** or **group** for which you want to make the device trusted.
User or group name	Name of a local or domain user (without specifying the full name with the domain, if the system allows entering user names in this format, or in the `<domain name>\<user name>` or `<user name>@<domain name>`) format, or the name of a group of users for which you want to make the device trusted.

Device type window

In this window, you can configure access mode for various types of devices.

Access mode for device types

Setting	Description
Device type	Device type (for example, Hard drives, Printers).
Access mode	Device access mode. Right-clicking with the mouse opens a context menu where you can select one of the following items: • **Allow**: allow access to devices of the selected type. • **Block**: prohibit access to devices of the selected type. • **Depends on bus** (default value): allow or block access to the devices depending on the access mode for a connection bus. • **By rule**: allow or block access to devices depending on the access rule and schedule. This menu command is available only for storage devices (hard drives, removable drives, floppy disk drives, CD/DVD drives). • **Read only**: allow read-only access to devices of the selected type. This menu command is available only for storage devices (hard drives, removable drives, floppy disk drives, CD/DVD drives).

You can configure access rules and schedules in the **Configure device access rule** window, which opens when you double-click the device type.

Configure device access rule window

In this window, you can configure access rules and schedules for the selected device type.

This window is opened by double-clicking the device type in the **Device type** window.

Device access rules and schedules

Setting	Description
Users and/or user groups	The list contains users and groups for which you can configure access schedule. By default, the table contains the **\Everyone** item (all users). You can add, edit, or delete users or user groups.
Device access rules	This table contains access schedules for users and user groups. It consists of the following columns: • **Access schedule** – names of existing access schedules. The check box next to the schedule indicates whether this schedule is used by the component. • **Access** – access type for the schedule: **Allow** (grant access to devices of the selected type) or **Block** (deny access to devices of the selected type). You can configure schedules only for hard drives, removable drives, floppy disks, and CD/DVD drives. By default, the table contains the **Default** access schedule, which provides all users with full access to devices (the **\Everyone** item is selected in the **Users and/or user groups** list) at any time if access via the connection bus is allowed for this type of device. You can add, edit, or delete access schedules for selected users. The **Default** schedule cannot be modified or removed.

User or group window

In this window, you can specify a user or group of users to which the device access rule applies.

Configure device access rule

Setting	Description
Type	The **User** or **Group** to which the Application Control rule applies.
User or group name	Name of a local or domain user (without specifying the full name with the domain, if the system allows entering user names in this format, or in the <`domain name`>\<`user name`> or <`user name`>@<`domain name`>) format, or the name of a group of users to which the rule applies.

Access schedule window

In this window, you can configure the device access schedule.

Schedule for access to devices

Setting	Description
Name	Entry field for the access schedule name.
Time intervals	The table where you can select time intervals for the schedule (days and hours). Intervals highlighted in green are included to the schedule. To exclude an interval from the schedule, click the corresponding cells. Intervals excluded from the schedule are highlighted in gray. By default, all intervals (24/7) are included to the schedule.

Connection buses window

In this window, you can configure access mode for connection buses.

Access mode for connection buses

Setting	Description
Connection bus	Connection bus used by devices to connect to the client device: • **FireWire** • **USB**
Access mode	Connection bus access mode. Right-clicking with the mouse opens a context menu where you can select one of the following items: • **Allow** (default): provide access to devices connected through this bus. • **Block**: deny access to devices connected using this connection bus.

Configuring the Device Control message template

When a user tries to gain access to a blocked device, the application displays a message about access denial or the blocking of an operation with the device.

Access denial messages are based on a template that describes the reasons for blocking. You can use special variables (for example,%USER_NAME%) in the message template. Such variables allow using the same message template for different devices. When generating a message, the application substitutes the values of variables corresponding to the specific device, user, and event into the message.

You can edit the message template.

To edit the message template in the Web Console:

1. In the policy properties window, select **Application settings** → **Security Controls** → **Device Control**.
2. In the **Message templates** section, click the **Block messages** link.
3. This opens the **Message templates** window; in that window, configure the Device Control message template on the **Message to user** tab.

   You can add variables and links to the body of the message for the user by clicking the **Add variable** and **Add link** buttons. To add a variable to the template, place the cursor in the required place in the text, click the **Add variable** button, and select a variable in the drop-down list. To add a link to the template, place the cursor in the required place in the text and click the **Add link** button.

   You can also restore the text of the message template by clicking the **Reset to default** button.
4. Click **OK**.
5. Click the **Save** button to save the changes made.

To change a message template in the Administration Console:

1. In the **Managed devices** folder in the Administration Console tree, open the folder with the name of the administration group to which the required devices belong.
2. In the workspace, select the **Policies** tab.
3. In the list of policies, select the required policy and double-click it to open the **Properties: <Policy name>** window.

   You can also open the policy properties window by using the **Properties** item in the policy context menu or by clicking the **Configure policy settings** link located to the right of the list of policies in the section with the policy settings.
4. In the policy window, select **Security Controls** → **Device Control**.
5. In the **Message templates** section, click the **Configure** button.
6. This opens the **Message templates** window; in that window, configure the Device Control message template on the **Block message** tab.

   You can add variables and links to the body of the message for the user by clicking the **Variable** and **Link** buttons. To add a variable to the template, place the cursor in the required place in the text, click the **Variable** button, and select a variable in the drop-down list. To add a link to the template, place the cursor in the required place in the text and click the **Link** button.

   You can also restore the text of the message template by clicking the **Default** button.
7. Click **OK**.
8. Click **Apply**.

Configuring Device Control on the command line

You can manage Device Control in the command line with the help of the Device Control predefined task (*Device_Control*).

The Device Control task is not started by default. You can start and stop the task manually.

You can configure Device Control by editing the settings of the Device Control predefined task.

You can also use Device Control management commands to:

- view the list of connected devices and export it to file
- view and change the list of trusted devices

Device Control task settings

The table describes all possible values and the default values of all the settings that you can specify for the Device Control task.

Device Control task settings

Setting	Description	Values
`OperationMode`	Response to attempts to access a device that is restricted according to Device Control rules.	`Block` (default value) – the application applies the access mode defined for the device or bus. `Notify` – the application tests the selected access mode and generates an event about the detection of an attempt to access a device.
`AllowToRequestTemporaryAccess`	Allow requesting temporary access to a blocked device. Granting access to a blocked device is available only if Kaspersky Security Center 16.2 is deployed in the organization and the client device is able to establish a connection with the Administration Server.	`Yes` – allow the user to request temporary access to the blocked device in the graphical interface of the application or on the command line. `No` (default) – do not allow the user to request temporary access to a blocked device.
The **[DeviceClass]** section contains access modes for devices depending on their type.		
`HardDrive`	Access mode for the hard drives connected to a client device.	`Allow` — Users are allowed access to hard drives. `DependsOnBus` (default): access to the hard drive depends on the access mode defined for the bus through which it is connected. `Block` — Access to all hard drives (except system hard drives, which are never blocked by the Device Control) is blocked for users. `ByRule` — Access to the hard drives depends on the access rules. `ReadOnly` — Users have read only access to hard drives, writing is blocked.

Setting	Description	Values
`RemovableDrive`	Access mode for the removable drives connected to a client device.	`Allow` — Access to the removable drives is allowed for users. `DependsOnBus` (default): access to the removable drive depends on the access mode defined for the bus through which it is connected. `Block` — Access to the removable drives is blocked for users. `ByRule` — Access to the removable drives depends on the access rules. `ReadOnly` — Users have read only access to removable drives, writing is blocked.
`FloppyDrive`	Access mode for the floppy disks connected to a client device. The application does not block floppy disks connected to the client device using the ISA bus.	`Allow` — Users are allowed access to floppy disks. `DependsOnBus` (default): access to the floppy disk depends on the access mode defined for the bus through which it is connected. `Block` — Access to floppy disks is blocked for users. `ByRule` — Access to floppy disks depends on the access rules. `ReadOnly` — Users have read only access to floppy disks, writing is blocked.
`OpticalDrive`	Access mode for the CD/DVD drives connected to a client device.	`Allow` — Users are allowed access to CD/DVD drives. `DependsOnBus` (default): access to the CD/DVD drive depends on the access mode defined for the bus through which it is connected. `Block` — Access to CD/DVD drives is blocked for users. `ByRule` — Access to CD/DVD drives depends on the access rules. `ReadOnly` — Users have read only access to CD/DVD drives, writing is blocked.
`SerialPortDevice`	Access mode for the devices connected to a client device via a serial port. The application does not block the devices connected to a client device via a serial port using the ISA bus.	`Allow` — Users are allowed access to devices connected through a serial port. `DependsOnBus` (default): access to a device connected through a serial port depends on bus access mode. `Block` — Access to devices connected through a serial port is blocked for users.
`ParallelPortDevice`	Access mode for the devices connected to a client device via a parallel port.	`Allow` — Users are allowed access to devices connected through a parallel port. `DependsOnBus` (default): access to a device connected through a parallel port depends on bus access mode. `Block` — Access to devices connected through a parallel port is blocked for users.
`Printer`	Access mode for the printers connected to a client device.	`Allow` — Users are allowed access to printers. `DependsOnBus` (default): access to a printer depends on the access mode defined for the bus through which it is connected. `Block` — Access to printers is blocked for users.
`Modem`	Access mode for the modems connected to a client device.	`Allow` — Users are allowed access to modems. `DependsOnBus` (default): access to a modem depends on the access mode defined for the bus through which it is connected. `Block` — Access to modems is blocked for users.
`TapeDrive`	Access mode for the tape devices connected to a client device.	`Allow` — Users are allowed access to tape devices. `DependsOnBus` (default): access to a tape device depends on the access mode defined for the bus through which it is connected `Block` — Access to tape devices is blocked for users.

Setting	Description	Values
`MultifuncDevice`	Access mode for the multifunctional devices connected to a client device.	`Allow` — Users are allowed access to multifunctional devices. `DependsOnBus` (default): access to a multifunctional device depends on the access mode defined for the bus through which it is connected. `Block` — Access to multifunctional devices is blocked for users.
`SmartCardReader`	Access mode for the smart card readers connected to a client device.	`Allow` — Access to smart card readers is allowed for users. `DependsOnBus` (default): access to a smart card reader depends on the access mode defined for the bus through which it is connected. `Block` — Access to smart card readers is blocked for users.
`WiFiAdapter`	Access mode for the Wi-Fi adapters connected to a client device.	`Allow` — Users are allowed access to Wi-Fi adapters. `DependsOnBus` (default): access to a Wi-Fi adapter depends on connection bus access mode. `Block` — Access to the Wi-Fi adapters is blocked for users.
`NetworkAdapter`	Access mode for the external network adapters connected to a client device.	`Allow` — Users are allowed access to external network adapters. `DependsOnBus` (default): access to an external network adapter depends on the access mode defined for the bus through which it is connected. Device Control does not allow denying access to external network adapters to avoid disconnecting the client device from the network.
`PortableDevice`	Access mode for the portable devices connected to a client device.	`Allow` — Users are allowed access to portable devices. `DependsOnBus` (default): access to a portable device depends on the access mode defined for the bus through which it is connected. `Block` — Access to portable devices is blocked for users.
`BluetoothDevice`	Access mode for the Bluetooth devices connected to a client device.	`Allow` — Users are allowed access to Bluetooth devices. `DependsOnBus` (default): access to a Bluetooth device depends on the access mode defined for the bus through which it is connected. `Block` — Access to Bluetooth devices is blocked for users.
`ImagingDevice`	Access mode for the imaging devices connected to a client device.	`Allow`—Access to all imaging devices is allowed for users. `DependsOnBus` (default): access to an imaging device depends on the access mode defined for the bus through which it is connected. `Block`—Access to all imaging devices is blocked for users.
`SoundAdapter`	Access mode for the sound adapters connected to a client device.	`Allow`—Access to all sound adapters is allowed for users. `DependsOnBus` (default): access to a sound adapter depends on the access mode defined for the bus through which it is connected. `Block`—Access to all sound adapters is blocked for users.
`InputDevice`	Access mode for the input devices (keyboards, mouse, touchpad, and others) connected to a client device.	`Allow` — Users are allowed access to input devices. `DependsOnBus` (default): access to an input device depends on the access mode defined for the bus through which it is connected. `Block` — Access to input devices is blocked for users.
The **[DeviceBus]** section contains access modes for connection buses.		
`USB`	Access mode for devices connected to the client device via USB.	`Allow` (default value) — Users are allowed access to USB-devices. `Block` — Access to USB-devices is blocked for users.

Setting	Description	Values
`FireWire`	Access mode for devices connected to the client device via FireWire.	`Allow` (default value) — Users are allowed access to devices connected via the FireWire interface. `Block` — Access to devices connected via the FireWire interface is blocked for users.
The **[TrustedDevices.item_ #]** section contains trusted devices.		
`DeviceId`	Specifies ID or ID mask of a trusted device.	You can use the masks * (any sequence of characters) or ? (any single character) to indicate the device ID. Examples: *To deny access to all USB devices except the specified one, specify the following settings:* In the `[DeviceBus]` section, specify `USB=Block` In the `[TrustedDevices.item_#]` section, specify `DeviceId=<device ID>` *To deny access to all USB devices, but allow access to all USB drives, specify the following settings:* In the `[DeviceBus]` section, specify `USB=Block` In the `[TrustedDevices.item_#]` section, specify `DeviceId=USBSTOR*`
`Comment`	Comment to the specified trusted device.	—
The **[TrustedDevices.item_#.Principals.item_#]** section contains the rules for accessing a trusted device.		
`Principal`	The user or group for which the device is trusted. If the device has the "read-only" access type, when the trusted device is assigned to a specific user, the assigned user is ignored and the device is made available for writing to all users.	`\Everyone` (default): the device is trusted for all users. `<user name>` is the name of a local or domain user for which the device is trusted (without specifying the full name with the domain, if the system allows entering user names in this format, or in the `<domain name>\<user name>` or `<user name>@<domain name>`) format. `@<group name>` is the name of the (local or domain) group of users for which the device is trusted.
The **[Schedules.item_#]** section contains the device access schedule. You can configure a schedule only for hard drives, removable drives, floppy disks, and CD/DVD drives.		
`ScheduleName`	Specifies a schedule name. The schedule name must be unique.	Default value: `Default`. The `Default` schedule provides users full access to devices at any time if the connection bus is allowed to access the corresponding device type. You cannot delete the `Default` schedule.

Setting	Description	Values
DaysHours	Specifies time intervals for a schedule.	All (default value) — The schedule is valid 24/7 (no time limitation). < weekday > is the day of the week. You can use either the full week day names or abbreviations (for example, for Monday, you can specify Mo, or Mon , or Monday). For week days, you can specify intervals or specific days. The week starts from Sunday. < hour > is the hours [0:24]. For hours, you can specify only intervals. Examples: *Schedule_1 is valid from Sunday till Saturday from 0 a.m. to 11 a.m., from 12 p.m. to 3 p.m., and from 4 p.m. to 12 a.m.:* [Schedules.item_0001] ScheduleName=schedule_1 DaysHours=Su-Sa:0..11,12..15,16..24 *Schedule_2 is valid for the following intervals: on Thursdays from 12 p.m. to 2 p.m. and on Fridays from 2 a.m. to 3 p.m. and from 4 p.m. to 12 a.m.:* [Schedules.item_0002] ScheduleName=schedule_2 DaysHours=Th:12..14;Fr:2..15,16..24 *Schedule_3 is valid 24 hours 7 days a week:* [Schedules.item_0003] ScheduleName=schedule_3 DaysHours=All
The **[HardDrivePrincipals.item_#]** section contains hard drive access rules. For hard drives, at least one schedule must always be enabled. You can assign several access rules to a hard drive. Also, multiple schedules can be specified for a user or group of users. If an access rule conflict occurs for a user or group, the minimum access rights are granted.		
Principal	Specifies a user or group of users for whom the access rule is applied.	\Everyone (default value) — The access rule applies to all users. < user name > is the name of a local or domain user to which the access rule applies (without specifying the full name with the domain, if the system allows entering user names in this format, or in the <domain name>\<user name> or <user name>@<domain name> format). @< group name > is the name of the (local or domain) group of users to which the access rule applies.
[HardDrivePrincipals.item_#.AccessRules.item_#] contains access rule settings.		
UseRule	Specifies whether the rule is enabled or disabled.	Yes (default value) — The access rule is enabled. No — The access rule is disabled.
ScheduleName	Schedule specified in the [Schedules.item_#] section.	Default value: Default.
Access	Specifies access type.	Allow (default value) — Access to hard drives is allowed. Block — Access to hard drives is blocked.
The **[RemovableDrivePrincipals.item_#]** section contains the access rules for removable drives. For removable drives, at least one schedule must always be enabled. You can assign several access rules to a removable drive. Also, multiple schedules can be specified for a user or group of users. If an access rule conflict occurs for a user or group, the minimum access rights are granted.		
Principal	Specifies a user or group of users for whom the access rule is applied.	\Everyone (default value) — The access rule applies to all users. < user name > is the name of a local or domain user to which the access rule applies (without specifying the full name with the domain, if the system allows entering user names in this format, or in the <domain name>\<user name> or <user name>@<domain name> format). @< group name > is the name of the (local or domain) group of users to which the access rule applies.
[RemovableDrivePrincipals.item_#.AccessRules.item_#] contains access rule settings.		

Setting	Description	Values
UseRule	Specifies whether the rule is enabled or disabled.	Yes (default value) — The access rule is enabled. No — The access rule is disabled.
ScheduleName	Schedule specified in the [Schedules.item_#] section.	Default value: Default.
Access	Specifies access type.	Allow (default value) — Access to removable drives is allowed. Block — Access to removable drives is blocked.
The **[FloppyDrivePrincipals.item_#]** section contains access rules for floppy drives. For floppy drives, at least one schedule must always be enabled. You can assign several access rules to a floppy drive. Also, multiple schedules can be specified for a user or group of users. If an access rule conflict occurs for a user or group, the minimum access rights are granted.		
Principal	Specifies a user or group of users for whom the access rule is applied.	\Everyone (default value) — The access rule applies to all users. < user name > is the name of a local or domain user to which the access rule applies (without specifying the full name with the domain, if the system allows entering user names in this format, or in the <domain name>\<user name> or <user name>@<domain name> format). @< group name > is the name of the (local or domain) group of users to which the access rule applies.
[FloppyDrivePrincipals.item_#.AccessRules.item_#] contains access rule settings.		
UseRule	Specifies whether the rule is enabled or disabled.	Yes (default value) — The access rule is enabled. No — The access rule is disabled.
ScheduleName	Schedule specified in the [Schedules.item_#] section.	Default value: Default.
Access	Specifies access type.	Allow (default value) — Access to floppy drives is allowed. Block — Access to floppy drives is blocked.
The **[OpticalDrivePrincipals.item_#]** section contains the access rules for CD/DVD drives. For CD/DVD drives, at least one schedule must always be enabled. You can assign several access rules to a CD/DVD drive. Also, multiple schedules can be specified for a user or group of users. If an access rule conflict occurs for a user or group, the minimum access rights are granted.		
Principal	Specifies a user or group of users for whom the access rule is applied.	\Everyone (default value) — The access rule applies to all users. < user name > is the name of a local or domain user to which the access rule applies (without specifying the full name with the domain, if the system allows entering user names in this format, or in the <domain name>\<user name> or <user name>@<domain name> format). @< group name > is the name of the (local or domain) group of users to which the access rule applies.
[OpticalDrivePrincipals.item_#.AccessRules.item_#] contains access rule settings.		
UseRule	Specifies whether the rule is enabled or disabled.	Yes (default value) — The access rule is enabled. No — The access rule is disabled.
ScheduleName	Schedule specified in the [Schedules.item_#] section.	Default value: Default.
Access	Specifies access type.	Allow (default value) — Access to CD/DVD drives is allowed. Block — Access to CD/DVD drives is blocked.

Viewing and exporting the list of connected devices on the command line

To view the list of connected devices, execute the following command:

```
kess-control --get-device-list [--all] [--json]
```

where:

- `--all` lists all connected devices.
- `--json` lists connected devices in JSON format.

Kaspersky Embedded Systems Security displays the following information about connected devices:

- **Device type**. Type of the connected device. For example, `OpticalDrive` or `HardDrive`.
- **Identifier**. ID of the connected device.
- **Name**. Name of the connected device.
- **Path**. Path to the device in the sysfs virtual operating system.
- **System drive**. The setting indicates whether the connected device is a system drive (Yes or No).
- **Bus**. Connection bus. Possible values: `UnknownBus`, `USB`, `FireWire`.
- **Driver**. Name of the driver read by the sysfs virtual operating system.

To export the list of connected devices, run the following command:

```
kess-control --get-device-list --export <file>
```

where `<file>` is the full path to the JSON file in which you want to save the list of connected devices.

The list of trusted devices is exported.

Managing the list of trusted devices on the command line

The list of trusted devices contains information about devices that are allowed to be accessed (for specific users or groups).

In the command line, you can view the list of trusted devices, add and remove devices from the list, and import the list of trusted devices from a file.

To view the list of trusted devices, run the following command:

```
kess-control --trusted-devices --list
```

To add a device to the list of trusted devices, run the following command:

```
kess-control --trusted-devices --add <ID>
```

where `<identifier>` is the device ID or device ID mask.

To remove a device from the list of trusted devices, run the following command:

```
kess-control --trusted-devices --delete <ID>
```

where `<identifier>` is the device ID or device ID mask.

To import a list of trusted devices from a file, run the following command:

```
kess-control --trusted-devices --merge|--add|--replace --file <file>
```

where:

- `--merge` merges the lists of trusted devices. If the current list and the imported list contain entries for a device with the same ID, the existing entries are updated. For devices that are not in the current list, entries are added.
- `--add` adds the current list of trusted devices when importing. Only entries for devices not previously on the list of devices are added to the list.
- `--replace` replaces the current list of trusted devices. The existing list of trusted devices is replaced with the imported list.
- `--file <file>` is the full path to the JSON file that the list of trusted devices will be imported from.

Granting temporary access to device

When configuring Device Control, the administrator may mistakenly block access to a device that the user needs for work. If the user believes that access to a device or an operation with device content was blocked by mistake, the user can request temporary access to the blocked device from the administrator of the corporate local network.

The user can request temporary access if the administrator has the user allowed to do so. The administrator can configure this permission in the policy properties in the Web Console, in the Administration Console, or in the settings of the Device Control task).

Granting access to a blocked device involves the following steps:

1. The user creates an *access request file* and sends it to the administrator, for example, by email or in any available way (the administrator can configure this in the block message template). The user can request temporary access to the blocked device in the graphical interface of the application or on the command line.

2. The administrator creates an *access key* based on the access request file and passes it to the user, for example, by email or in any other way. Access to a blocked device is granted in the Web Console.

3. The user uploads the access key and gains temporary access to the blocked device. The user can upload the access key in the graphical user interface or on the command line of the application.

Granting access to a blocked device is available only if Kaspersky Security Center 16.2 is deployed in the organization and the client device is able to establish a connection with the Administration Server.

Granting access to a blocked device in the Web Console

Granting access to a blocked device is available only if Kaspersky Security Center 16.2 is deployed in the organization and the client device is able to establish a connection with the Administration Server.

When the administrator receives a temporary access request from a user, the administrator can create an access key from the access request file in the Web Console.

To provide temporary access to the device, the administrator must have read, write, and execute permissions for the **Data Encryption** function (**Users & roles** → **Roles** → <user role> → user role properties window → **Access rights** tab → <application name> → **Data Encryption** → **Read/Write/Execute**).

To create an access key for a blocked device:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select the administration group containing the relevant device and select the check box next to the name of the device.

3. Above the list of managed devices, click the **Grant access to the device in offline mode** button. If this button is not displayed above the list of devices, click the three dots button and select the **Grant access to the device in offline mode** command from the menu.

4. This opens the **Application Settings** window; in that window, in the **Device Control** section, click the **Browse** button and download the access request file received from the user.

   Information about the blocked device to which the user has requested access is displayed.

5. If necessary, change the duration in hours in the **Access duration** field.

   By default, **Access duration** is set to the value specified by the user when generating the access request file.

6. If necessary, change the value in the **Specify how long an access key can be used on the client device (in hours)** field.

   This field contains the time period during which the access key can be uploaded by the user.

7. Click the **Create key** button.

8. Save the access key file with the *.acode extension.

Gaining temporary access to the device on the command line

If the user believes that access to the device was blocked or an operation with the contents of the device was prohibited by mistake, the user can request temporary access to the blocked device from the administrator of the local network of the organization using the following Device Control managementcommand: `kess-control --request-temporary-device-access`.

To request temporary access to a blocked device, run the following command:

```
kess-control --request-temporary-device-access --device-id <device ID> --access-duration <access duration> --path <path to access request file>
```

where:

`--device-id <device ID>` specifies the ID of the blocked device to which you want to gain access.

`--access-duration <access duration>` specifies the time period during which you want to be able to gain access to the device (in hours). Possible values: 1 to 100

`--path <path to the access request file>` specifies the path to the directory in which you want to save the access request file with the .akey extension.

This command creates an access request file, which must be sent to the administrator, for example, by email or in any other way (the administrator can describe the preferred method in the template of the blocking message).

After getting an access key from the administrator, you need to upload the access key using the command `kess-control --upload-temporary-device-access-key` to gain temporary access to the device.

To gain temporary access to a blocked device, run the following command:

```
kess-control --upload-temporary-device-access-key --path <path to the file with the access key>
```

where:

`--path <path to the file with the access key>` is the path to the file containing the access key to the device (file with the .acode extension).

This command outputs information about the granted access to the device and the access duration.

System Integrity Monitoring

Kaspersky Embedded Systems Security monitors the integrity of the operating system on the protected device in real time or on demand.

- *System Integrity Monitoring* tracks in real time changes in files and directories that you have added to the monitoring scope in the component settings. You can track changes in files that may indicate a security breach on a protected device.
- You can use the *System Integrity Check* tasks to check for changes in files and directories that you have added to the monitoring scope by comparing the current state of a monitored object with a previously recorded state.

Using System Integrity Monitoring requires a license that includes this feature.

Upon detecting changes to files or directories in the monitoring scope, Kaspersky Embedded Systems Security generates events about changes in object access control lists. System Integrity Monitoring does not share data on exact changes that were made. The *System Integrity Check* task sends data on modified attributes, and moved files and directories.

Real-time System Integrity Monitoring

System Integrity Monitoring detects each change to an object within the monitoring scope by intercepting file operations in real time.

When System Integrity Monitoring runs, the application monitors changes in the following file settings:

- Content (write (), truncate (), etc.)
- Metadata (possession rights (chmod/chown))
- Time stamps (utimensat)
- Extended attributes ((setxattr) and others)

A file checksum is not calculated.

The technical limitations of the Linux operating system prevent the application from identifying the user or process that made the changes to the file.

System Integrity Monitoring is disabled by default. You can enable, disable, and configure System Integrity Monitoring:

- Define monitoring scopes for System Integrity Monitoring The application monitors operations on files within the monitoring scopes defined in the System Integrity Monitoring settings. You have to specify at least one monitoring scope for the component to work. The *Kaspersky internal objects* (/opt/kaspersky/kess/) monitoring scope is defined by default.

  You can specify several monitoring scopes. You can change monitoring scopes in real-time mode.

The application task does not monitor changes in files (attributes and content) with hard links that are outside the monitoring scope.

- You can configure exclusion of objects from monitoring with the help of name masks.
- Set up exclusion scopes for System Integrity Monitoring. Exclusions are defined for each individual monitoring scope and only work for the indicated scope. You can specify several monitoring exclusions.

  An exclusion has a higher priority than a monitoring scope; an excluded object is skipped even if within the monitoring scope. If the monitoring scope is defined on a lower level than the excluded directory, the application skips this monitoring scope during system integrity monitoring.

When a directory is added to a monitoring or exclusion scope, the application does not check whether that directory exists.

Configuring System Integrity Monitoring in the Web Console

In the Web Console, you can configure System Integrity Monitoring settings in the policy properties (**Application settings** → **Security Controls** → **System Integrity Monitoring**).

System Integrity Monitoring settings

Setting	Description
Monitoring scopes	The table contains monitoring scopes of the System Integrity Monitoring component. The application monitors files and directories located in the paths specified in the table. By default, the table contains the *Kaspersky internal objects* monitoring scope.
Exclusion scopes	This table contains scan exclusions scopes. The application does not scan files and directories located at the paths specified in the table. By default, the table is empty.
Exclusions by mask	You can configure scan exclusions for objects by name mask. The application will not scan files whose names contain the specified mask. By default, the list of masks is empty.

Configuring System Integrity Monitoring in the Administration Console

In the Administration Console, you can manage System Integrity Monitoring settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls** → **System Integrity Monitoring**).

System Integrity Monitoring settings

Setting	Description
Enable System Integrity Monitoring	This check box enables or disables System Integrity Monitoring. This check box is cleared by default.
Monitoring scopes	The group of settings contains the **Configure** button. Clicking this button opens the **Scan scopes** window.
Monitoring exclusions	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window.

Scan scopes window

The table contains monitoring scopes for the System Integrity Monitoring component. The application monitors files and directories located in the paths specified in the table. By default, the table contains one monitoring scope, **Kaspersky internal objects** (/opt/kaspersky/kess/).

Monitoring scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Scope name	Monitoring scope name.
Path	Path to the directory that the application protects.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes, in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure monitoring scopes for the System Integrity Monitoring component.

Monitoring scope settings

Setting	Description
Scan scope name	Field for entering the monitoring scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application controls this monitoring scope during the application's operation. If this check box is cleared, the application does not control this monitoring scope during the operation. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to include in the monitoring scope. The field must not be blank. The default path is /opt/kaspersky/kess.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusion scopes window

The window contains monitoring exclusion scopes for the System Integrity Monitoring component. The application does not scan files and directories located at the paths specified in the table.

Monitoring exclusion scope settings

Setting	Description
Merge inherited values	This check box enables or disables the merging of the list when inherited in child policies. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from monitoring.
Status	Indicates whether the application excludes this scope from monitoring during the component operation.

By default, the table of exclusion scopes is empty. You can add, edit, or delete items in the table.

You can also import the list of exclusion scopes from a file by clicking **Import** or export the list of exclusion scopes to a file in JSON format by clicking **Export**. When importing, you are prompted to replace the list of exclusion scopes or add exclusion scopes to the existing list.

<New exclusion scope> window

In this window, you can add or configure the monitoring exclusion scope for the System Integrity Monitoring component.

Monitoring exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables the exclusion of the scope from monitoring when the application is running. If this check box is selected, the application excludes this scope from monitoring during the component operation. If this check box is cleared, the application monitors this scope during the component operation. You can later exclude this scope from monitoring by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to add to the exclusion scope. The field must not be blank. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Masks	The list contains name masks of the objects that the application excludes from the monitoring. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks. Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from monitoring. Examples: The `*.txt` mask refers to all text files. The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

Exclusions by mask window

You can configure the exclusion of objects from monitoring based on name masks. The application does not scan the files with the names containing the specified masks. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from monitoring.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Configuring System Integrity Monitoring on the command line

You can manage system integrity monitoring in real time in the command line by using the System Integrity Monitoring predefined task (*System_Integrity_Monitoring*). Task type: *OAFIM*.

The System Integrity Monitoring task does not run by default. You can start and stop the task manually.

You can configure System Integrity Monitoring on the device by editing the settings of the System Integrity Monitoring predefined task.

On-access System Integrity Monitoring task settings

Setting	Description	Values
`UseExcludeMasks`	Enables monitoring scope exclusions for objects specified by the `ExcludeThreats.item_#` setting. This setting applies only if the `ExcludeMasks.item_#` setting is specified.	`Yes` — Exclude the objects specified by the `ExcludeMasks.item_#` setting from the monitoring scope. `No` (default value) — Do not exclude the objects specified by the `ExcludeMasks.item_#` setting from the monitoring scope.
`ExcludeMasks.item_#`	Excludes objects from monitoring by names or masks. You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format. Before specifying a value for this setting, make sure that the `UseExcludeMasks` setting is enabled. You can specify several masks. Each mask must be specified on a new line with a new index.	The default value is not defined.
The **[ScanScope.item_#]** section contains the monitoring scopes of the System Integrity Monitoring task. At least one monitoring scope must be specified for the task. You can specify several [ScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of monitoring scope; contains additional information about the monitoring scope.	The default value is not defined.
`UseScanArea`	Enables monitoring of the specified scope.	`Yes` (default value) — Monitor the specified scope. `No` — Do not monitor the specified scope.

Setting	Description	Values
`Path`	Path to the monitoring directory.	You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. Default value: /opt/kaspersky/kess/
`AreaMask.item_#`	Monitoring scope limitation. Within the monitoring scope, the application scans only the objects that are specified using the masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (all objects are monitored)
[ExcludedFromScanScope.item_#] contains objects to be excluded from all [ScanScope.item_#] sections. You can specify multiple [ExcludedFromScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ExcludedFromScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of the monitoring exclusion scope, which contains additional information about the monitoring exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from monitoring.	`Yes` (default value) — Exclude the specified scope from monitoring. `No` — Do not exclude the specified scope from monitoring.
`Path`	Path to the directory with objects excluded from monitoring.	You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The default value is not defined.
`AreaMask.item_#`	Limitation of monitoring exclusion scope. In the monitoring exclusion scope, the application only excludes the objects that are specified using masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (exclude all objects from monitoring)

System Integrity Check

When the *System Integrity Check* task is running, a change in each object is found by comparing the current state of the monitored object with its original state. The following comparison criteria can be used:

- File hash
- File change time
- File size

The initial state of monitored objects is recorded as a *baseline*. The baseline contains paths to monitored objects and their metadata.

A baseline may also contain personal data.

A system baseline is created when a System Integrity Check task runs on the device for the first time. If you have created multiple System Integrity Check tasks, a separate baseline is created for each. The task is only executed if the baseline contains information about objects that belong to the monitoring scope defined for the task. If the baseline does not match the monitoring scope, Kaspersky Embedded Systems Security generates a system integrity violation event.

The baseline is updated whenever the task settings are modified (for example, when a new monitoring scope is added), as well as when the task is started with `RebuildBaseline=Yes` setting (or if the **Rebuild baseline on each task start** check box is selected in the task properties in Kaspersky Security Center).

The application creates a baseline storage on the protected device. By default, the storage for baselines is located in /var/opt/kaspersky/kess/private/fim.db. Root privileges are required to access a database that contains baselines.

You can delete a baseline by deleting the appropriate System Integrity Check task.

You can run a system integrity check on demand and configure the scan settings:

- Enable or disable baseline rebuild every time a system integrity check task finishes.
- Select criteria for comparing the current state of the monitored file with the original state: use the file hash and change time, or only the file size.
- Configure monitoring scopes for checking system integrity.
- Configure exclusion scopes from the system integrity check. You can specify paths to excluded files and directories, and exclude individual objects by name mask.

Configuring System Integrity Check in the Web Console

You can run a system integrity check in the Web Console with the help of the *System Integrity Check* task.

You can create and run user system integrity check tasks. You can configure the scan settings by editing the settings of the tasks.

System Integrity Check task settings

Setting	Description
Rebuild baseline on each task start	This check box enables or disables the reestablishment of a system baseline upon every start of the *System Integrity Check* task. This check box is cleared by default.
Check SHA256 hash	This check box enables or disables the use of the file hash as a criterion when comparing the current state of the file with its original state. If this check box is cleared, the application compares only the file size (if the file size has not changed, then the modification time is not considered a critical parameter). This check box is cleared by default.
Track directories in monitoring scopes	This check box enables or disables directory monitoring while system integrity check is running. This check box is cleared by default.
Track the last time a file was accessed	This check box enables or disables tracking the file access time while the system integrity check runs. This check box is cleared by default.
Monitoring scopes	The table that contains the monitoring scopes scanned by the task. By default, the table contains the **Kaspersky internal objects** (/opt/kaspersky/kess/) monitoring scope. You can add, edit, delete, move up, or move down monitoring scopes in the table. You can also import the list of monitoring scopes from a file by clicking **Import** or export the list of monitoring scopes to a file in JSON format by clicking **Export**. When importing, you are prompted to replace the list of monitoring scopes or add monitoring scopes to the existing list.

Add scan scope window

In this window, you can add or configure the monitoring scope for the System Integrity Check task.

Monitoring scope settings

Setting	Description
Scope name	Field for entering the monitoring scope name. This name will be displayed in the table in the **Scan settings** section. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application controls this monitoring scope during the application's operation. If this check box is cleared, the application does not control this monitoring scope during the operation. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to include in the monitoring scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank. The **/** path is specified by default – the application scans all directories of the local file system.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusion scopes section

In the **Exclusion scopes** section for the System Integrity Check task, you can also configure exclusion scopes for the scan and exclusions by mask.

Exclusion scopes window

The table contains monitoring exclusion scopes for the System Integrity Check component. The application does not scan files and directories located at the paths specified in the table. By default, the table is empty.

Monitoring exclusion scope settings

Setting	Description
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from monitoring.
Status	Indicates whether the application excludes this scope from monitoring during the task operation.

You can add, edit, or delete items in the table.

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

Add exclusion scope window

In this window, you can add and configure the monitoring exclusion scope for the System Integrity Check task.

Monitoring exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables the exclusion of the scope from monitoring when the application is running. If this check box is selected, the application excludes this scope from monitoring during the task operation. If this check box is cleared, the application monitors this scope during the task operation. You can later exclude this scope from monitoring by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank. The **/** path is specified by default. The application excludes all directories of the local file system from scan.

Setting	Description
Masks	The list contains name masks of the objects that the application excludes from the monitoring. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusions by mask window

You can configure the exclusion of objects from monitoring based on name masks. The application does not scan the files with the names containing the specified masks. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Configuring System Integrity Check in the Administration Console

You can perform a system integrity check in the Administration Console, with the help of the *System Integrity Check* task.

You can create and run user system integrity check tasks. You can configure the scan settings by editing the settings of the tasks.

In the **Settings** section of the properties of the System Integrity Check task, you can edit the settings listed in the table below.

System Integrity Check task settings

Setting	Description
Rebuild baseline on each task start	This check box enables or disables the rebuilding of the system baseline every time the System Integrity Check task is started. This check box is cleared by default.
Check SHA256 hash	This check box enables or disables the use of the file hash as a criterion when comparing the current state of the file with its original state. If this check box is cleared, the application compares only the file size (if the file size has not changed, then the modification time is not considered a critical parameter). This check box is cleared by default.

Setting	Description
Track directories in monitoring scopes	This check box enables or disables scanning of directories within the specified monitoring scopes during a system integrity check. This check box is cleared by default.
Track the last time a file was accessed	This check box enables or disables tracking the file access time while the system integrity check runs. This check box is cleared by default.
Monitoring scopes	The group of settings contains the **Configure** button. Clicking this button opens the **Scan scopes** window.

Under **Exclusion scopes** in the properties of the System Integrity Check, you can define monitoring exclusions and exclusions by mask.

Scan scopes window

The table contains monitoring scopes for the System Integrity Check task. The application monitors files and directories located in the paths specified in the table. By default, the table contains one monitoring scope, **Kaspersky internal objects** (/opt/kaspersky/kess/).

Monitoring scope settings

Setting	Description
Scope name	Monitoring scope name.
Path	Path to the directory that the application protects.
Status	The status indicates whether the application scans this scope.

You can add, edit, delete, move up, and move down items in the table.

Kaspersky Embedded Systems Security scans objects in the specified scopes, in the order they appear in the list of scopes. If necessary, place the subdirectory higher in the list than its parent directory, to configure security settings for a subdirectory that are different from the security settings of the parent directory.

<New scan scope> window

In this window, you can add and configure monitoring scopes for the System Integrity Check task.

Monitoring scope settings

Setting	Description
Scan scope name	Field for entering the monitoring scope name. This name will be displayed in the table in the **Scan scopes** window. The entry field must not be blank.
Use this scope	This check box enables or disables scans of this scope by the application. If this check box is selected, the application controls this monitoring scope during the application's operation. If this check box is cleared, the application does not control this monitoring scope during the operation. You can later include this scope in the component settings by selecting the check box. This check box is selected by default.

Setting	Description
File system, access protocol, and path	Entry field for the path to the local directory that you want to include in the monitoring scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank. The default path is /opt/kaspersky/kess.
Masks	The list contains name masks for the objects that the application scans. By default the list contains the * mask (all objects). You can add, edit, or delete masks.

Exclusion scopes section

Settings of scan exclusions

Group of settings	Description
Monitoring exclusions	This group of settings contains the **Configure** button. Clicking this button opens the **Exclusion scopes** window. In this window, you can define the list of scopes to be excluded from monitoring.
Exclusions by mask	This group of settings contains the **Configure** button, which opens the **Exclusions by mask** window. In this window, you can configure the exclusion of objects from monitoring by name mask.

Exclusion scopes window

The table contains scan exclusion scopes for the System Integrity Check component. The application does not scan files and directories located at the paths specified in the table. By default, the table is empty.

Scan exclusion scope settings for the System Integrity Check task

Setting	Description
Exclusion scope name	Exclusion scope name.
Path	Path to the directory excluded from scan.
Status	Indicates whether the application excludes this scope from monitoring during the component operation.

You can add, edit, or delete items in the table.

<New exclusion scope> window

In this window, you can add and configure the monitoring exclusion scope for the System Integrity Check task.

Monitoring exclusion scope settings

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in the table in the **Exclusion scopes** window. The entry field must not be blank.
Use this scope	The check box enables or disables the exclusion of the scope from monitoring when the application is running. If this check box is selected, the application excludes this scope from monitoring during the task operation. If this check box is cleared, the application monitors this scope during the task operation. You can later exclude this scope from monitoring by selecting the check box. This check box is selected by default.
File system, access protocol, and path	Entry field for the path to the local directory that you want to add to the exclusion scope. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The field must not be blank. The **/** path is specified by default. The application excludes all directories of the local file system from scan.
Masks	The list contains name masks of the objects that the application excludes from the monitoring. By default the list contains the ***** mask (all objects). You can add, edit, or delete masks. Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans. Examples: The `*.txt` mask refers to all text files. The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

Exclusions by mask window

You can configure the exclusion of objects from monitoring based on name masks. The application does not scan the files with the names containing the specified masks. By default, the list of masks is empty.

You can add, edit, or delete masks.

Clicking the **Add** button opens the **Object mask** window. In this window, in the **Define object mask** field, you can specify the name template for files that Kaspersky Embedded Systems Security excludes from scans.

Examples:
The `*.txt` mask refers to all text files.
The `*_my_file_??.html` mask refers to html files starting with any characters, and ending with `_my_file_` followed by any two characters (for example, 2020_my_file_09.html).

You can also import the exclusion list from a file using the **Import** button and export the exclusion list to a JSON file using the **Export** button. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

You can enable the merging of the list of exclusions when inherited by child policies. If you want to merge list items, select the **Merge inherited values** check box. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Configuring System Integrity Check on the command line

You can run a system integrity check on a device on the command line by using user *System Integrity Check* tasks (*ODFIM* tasks).

You can manually start, stop, pause, or resume user tasks and configure the task schedule. You can configure system integrity checking by editing the settings of these tasks.

System Integrity Check task settings

Setting	Description	Values
`RebuildBaseline`	Enables baseline to rebuild after the *System Integrity Check* task finishes.	`Yes`: rebuild the baseline every time the *System Integrity Check* task finishes. `No` (default): do not rebuild the baseline every time the *System Integrity Check* task finishes.
`CheckFileHash`	Use the file hash (SHA256) as a criterion when comparing the current state of the monitored file with its original state.	`Yes`: check the hash. `No` (default value) — Disable hash check. If this check is disabled, the application compares only the file size (if the file size has not changed, then the modification time is not considered a critical parameter).
`TrackDirectoryChanges`	Enables directory monitoring.	`Yes`: monitor directories while checking system integrity. `No` (default value) — Do not monitor directories.
`TrackLastAccessTime`	Enables tracking last file access time. In the Linux operating systems it is the `noatime` setting.	`Yes` — Track last file access time. `No` (default value) — Do not track last file access time.
`UseExcludeMasks`	Enables monitoring scope exclusions for objects specified by the `ExcludeMasks.item_#` setting. This setting applies only if the `ExcludeMasks.item_#` setting is specified.	`Yes` — Exclude the objects specified by the `ExcludeMasks.item_#` setting from the monitoring scope. `No` (default value) — Do not exclude the objects specified by the `ExcludeMasks.item_#` setting from the monitoring scope.
`ExcludeMasks.item_#`	Excludes objects from monitoring by names or masks. You can use this setting to exclude an individual file from the specified scan scope by name or exclude several files at once using masks in the shell format. Before specifying a value for this setting, make sure that the `UseExcludeMasks` setting is enabled. You can specify several masks. Each mask must be specified on a new line with a new index.	The default value is not defined.
The **[ScanScope.item_#]** section contains the monitoring scopes of the *System Integrity Check*. At least one monitoring scope must be specified for the task. You can specify several [ScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of monitoring scope; contains additional information about the monitoring scope.	The default value is not defined.
`UseScanArea`	Enables monitoring of the specified scope.	`Yes` (default value) — Monitor the specified scope. `No` — Do not monitor the specified scope.

Setting	Description	Values
`Path`	Path to the monitoring directory.	You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. Default value: /opt/kaspersky/kess/
`AreaMask.item_#`	Monitoring scope limitation. Within the monitoring scope, the application scans only the objects that are specified using the masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (all objects are monitored)
The **[ExcludedFromScanScope.item_#]** section contains the objects to be excluded from all [ScanScope.item_#] sections. You can specify several [ExcludedFromScanScope.item_#] sections in any order. The application processes the scopes by index in ascending order. The [ExcludedFromScanScope.item_#] section contains the following settings:		
`AreaDesc`	Description of the monitoring exclusion scope, which contains additional information about the monitoring exclusion scope.	The default value is not defined.
`UseScanArea`	Excludes the specified scope from monitoring.	`Yes` (default value) — Exclude the specified scope from monitoring. `No` — Do not exclude the specified scope from monitoring.
`Path`	Path to the directory with objects excluded from monitoring.	You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name. The default value is not defined.
`AreaMask.item_#`	Limitation of monitoring exclusion scope. In the monitoring exclusion scope, the application only excludes the objects that are specified using masks in the shell format. You can specify several `AreaMask.item_#` items in any order. The application processes the scopes by index in ascending order.	Default value: `*` (exclude all objects from monitoring)

Behavior Detection

The Behavior Detection component allows you to monitor for any malicious activity from applications in the operating system. When malicious activity is detected, Kaspersky Embedded Systems Security can terminate the process of the application that performs malicious activity.

The Behavior Detection component is enabled automatically with the default settings when Kaspersky Embedded Systems Security starts.

For optimal performance of the Behavior Detection component, we recommend enabling the Web Threat Protection component.

You can enable, disable, and configure Behavior Detection:

- Select an action to be performed by Kaspersky Embedded Systems Security upon detecting malicious activity in the operating system: inform the user or block the application that performs malicious activity.
- Exclude process activity from scans by the Behavior Detection component and the MDR component.

  The application automatically excludes the activity of the Watchdog server process from scanning. This exclusion cannot be removed.

If the root user does not have access to a remote directory mounted on the device over the NFSv4 protocol, the application blocks access to such a file for all users.

Configuring Behavior Detection in the Web Console

In the Web Console, you can manage Behavior Detection settings in the policy properties (**Application settings** → **Host Threat Protection** → **Behavior Analysis** → **Behavior Detection**).

Behavior Detection component settings

Setting	Description
Trusted applications	The *list of trusted applications* is a list of applications whose file and network activity Kaspersky Embedded Systems Security does not monitor (this includes malicious activity). By default, Kaspersky Embedded Systems Security monitors objects that are opened, launched, or saved by any software process, and also monitors the activity of all applications and the network traffic they generate. After an application is added to the list of trusted applications, Kaspersky Embedded Systems Security stops monitoring its activity.
Add application	You can add an application to the list of trusted applications by its path or path mask. Additionally, you can exclude the activity of child applications from scanning, as well as exclude an application from scanning by the Managed Detection and Response component.
Action on threat detection	**Inform**. If this action is selected, then upon detection of malicious activity, Kaspersky Embedded Systems Security does not terminate the process carrying out the malicious activity, but only logs an event about the detection of malicious activity in the event log. **Block**. If this action is selected, upon detection of malicious activity, Kaspersky Embedded Systems Security terminates the process carrying out the malicious activity and logs an event about the detected malicious activity to the event log.

Configuring Behavior Detection in the Administration Console

In the Administration Console, you can manage Behavior Detection settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Advanced Threat Protection → Behavior Detection**).

Behavior Detection component settings

Setting	Description
Enable Behavior Detection	This check box enables or disables the Behavior Detection component. This check box is selected by default.
Action on malware activity detection	The action to be performed by Kaspersky Embedded Systems Security upon detecting malicious activity in the operating system: • **Block** the application that performs malicious activity (default value). Kaspersky Embedded Systems Security terminates the process that performs malicious activity and logs information about the detected malicious activity. • **Inform** user. Kaspersky Embedded Systems Security does not terminate the process that performs malicious activity; it only records the detection of malicious activity in the event log.
Use exclusions by process	This check box enables or disables exclusions by process in the operation of the Behavior Detection component. This check box is cleared by default. The **Configure** button opens the **Exclusions by process** window. In this window, you can exclude the activity of processes.

Exclusions by process window

The table contains the exclusion scopes for exclusion by process The exclusion scope for exclusion by process lets you exclude the activity of an indicated process. By default, the table is empty.

Exclusion scope settings for exclusion by process

Setting	Description
Exclusion scope name	Exclusion scope name.
Path	Full path to excluded process.
Status	The status indicates whether the application uses this exclusion.

You can add, edit, or delete items in the table.

You can also import the list of exclusions from a file by clicking **Advanced** -> **Import** and export the list of added exclusions to a file by clicking **Advanced** -> **Export selected** or **Advanced** -> **Export all**. When importing, you will be prompted to replace the list of exclusions or add the exclusions to the existing list.

Trusted process window

In this window, you can add and configure exclusion scopes for exclusion by process.

Exclusion scope settings for exclusion by process

Setting	Description
Exclusion scope name	Field for entering the exclusion scope name. This name will be displayed in a table in the **Exclusions by process** window.
Path to excluded process	Full path to the process you want to exclude from scans. You can use **masks** to specify the path. You can use the * (asterisk) character to create a file or directory name mask. You can indicate a single * character to represent any set of characters (including an empty set) preceding the / character in the file or directory name. For example, /dir/*/file or /dir/*/*/file. You can indicate two consecutive * characters to represent any set of characters (including an empty set and the / character) in the file or directory name. For example, /dir/**/file*/ or /dir/file**/. The ** mask can be used only once in a directory name. For example, /dir/**/**/file is an incorrect mask. You can use a single ? character to represent any one character in the file or directory name. The entry field must not be blank.
Apply to child processes	Exclude child processes of the excluded process indicated by the **Path to excluded process** setting. This check box is cleared by default.
Use this scope	The check box enables or disables this exclusion scope. If this check box is selected, the application excludes this scope. If this check box is cleared, the application includes this scope. You can later exclude this scope by selecting the check box. This check box is selected by default.
Exclude process from Behavior Detection scans	This check box enables or disables the exclusion of the process from scanning by Behavior Detection. This check box is cleared by default.
Exclude process from MDR telemetry	This check box enables or disables the exclusion of the process from telemetry obtained by MDR. This check box is cleared by default.

Configuring Behavior Detection on the command line

You can manage application Behavior Detection in the operating system via the command line by using the *Behavior_Detection* predefined task.

The Behavior Detection task runs by default. You can start and stop the task manually.

You can configure Behavior Detection by editing the settings of the Behavior Detection predefined task.

Behavior Detection task setting

Setting	Description	Values
UseTrustedPrograms	Excluding processes from scans.	Yes – do not scan the activity of the indicated processes. No (default value) – scan all processes.
TaskMode	Action performed by the application when malicious activity is detected in the operating system.	Block (default value) – terminate the process of the application performing malicious activity. Notify – do not terminate the process performing malicious activity; only log detection of malicious activity in the event log.
The **[TrustedPrograms.item_#]** section contains processes that are excluded from Behavior Detection scans. Kaspersky Embedded Systems Security does not monitor the activity of the specified processes.		

Setting	Description	Values
`ProgramPath`	Path to excluded process.	`< full path to process >` – Do not scan the process in the specified local directory. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name.
`ApplyToDescendants`	Exclude child processes of the excluded process specified by the `ProgramPath` setting from scans.	`Yes` – exclude the specified process and all its child processes from scans. `No` (default value) – exclude only the specified process from scans, do not exclude its child processes from scans.
`ProgramDesc`	Description of the excluded process.	
`UseTrustedProgram`	Exclude the activity of the specified process from scanning.	`Yes` (default value) – Exclude the activity of the specified process from scanning. `No` – do not exclude the activity of the specified process from scanning.
`UseTrustedProgramForDetects`	Excluding the activity of a process from scanning by Behavior Detection.	If set to `Yes`, the activity of the specified process is excluded from scanning by Behavior Detection. If set to `No` (default), the activity of the specified process is not excluded from scanning by Behavior Detection.
`UseTrustedProgramForMDR`	Exclude process activity from telemetry obtained by the MDR component.	`Yes` – exclude the activity of the specified process from telemetry obtained by the MDR component. `No` (default value) – do not exclude the activity of this process from telemetry of the MDR component.
`TelemetrySource`	Additional setting for the obtaining of telemetry sent to the Managed Detection and Response solution when integrated with such.	`Default` (default) means the eBPF technology and auditd service are used to get telemetry. `OnlyEBPF` means that only the eBPF technology is used to get telemetry.
`UseTelemetryMonopolyMode`	Additional setting for the obtaining of telemetry sent to the Managed Detection and Response solution when integrated with such. The `UseTelemetryMonopolyMode` setting is applied only if `TelemetrySource` is set to `Default`.	`True` (default) means Kaspersky Embedded Systems Security uses the auditd service in exclusive mode. `False` means the auditd service is used in non-exclusive mode.

Using Kaspersky Security Network

The KSN functionality may not be available in the application in the territory of the USA.

To increase the protection of devices and user data, Kaspersky Embedded Systems Security can use Kaspersky's cloud-based knowledge base, Kaspersky Security Network (KSN), to check the reputation of files, Internet resources, and software. Using Kaspersky Security Network data ensures a faster response to various threats, high protection component performance, and fewer false positives.

Use of Kaspersky Security Network is voluntary. You can start or stop using KSN at any time.

Kaspersky Security Network infrastructure solutions

Kaspersky Embedded Systems Security supports the following infrastructure solutions to work with Kaspersky's reputation databases:

- *Kaspersky Security Network (KSN)* – A solution that receives information from Kaspersky and sends data about objects detected on user devices to Kaspersky for additional verification by Kaspersky analysts and to add to reputation and statistical databases.

- *Kaspersky Private Security Network (KPSN)* – A solution that allows users of devices with Kaspersky Embedded Systems Security installed to access the reputation databases of Kaspersky, as well as other statistical data, without sending data to Kaspersky from their devices. KPSN is designed for corporate clients who can't use Kaspersky Security Network, for example, for the following reasons:

  - No connection of local workplaces to the Internet

  - Legal prohibition or corporate security restrictions on sending any data outside the country or the organization's local network

To use KPSN after activating application under a new license, inform the service provider about the new license key. Otherwise, an authentication error will prevent data exchange with KPSN.

Kaspersky Security Network usage options:

There are two options for using KSN:

- **Extended KSN mode** – you can receive information from the Kaspersky knowledge base, while Kaspersky Embedded Systems Security automatically sends statistical information to Kaspersky Security Network that it obtained during its operation. The application can also send to Kaspersky for additional scanning certain files (or parts of files) that intruders can use to harm the device or data.

- **Basic KSN mode** – you can receive information from the Kaspersky knowledge base, but Kaspersky Embedded Systems Security does not send anonymous statistics and data about the types and sources of threats.

You can select a different Kaspersky Security Network usage option at any time.

No personal data is obtained, processed, or stored. Detailed information about the storage, and destruction, and/or submission to Kaspersky of statistical information generated during participation in KSN is available in the Kaspersky Security Network Statement and on Kaspersky's website. The file with the text of the Kaspersky Security Network Statement is included in the application distribution kit.

Cloud mode for Kaspersky Embedded Systems Security

Cloud mode is an operating mode of Kaspersky Embedded Systems Security that uses a lightweight version of the malware databases. This helps reduce the load on device memory.

Kaspersky Security Network facilitates the application's use of the lightweight anti-malware databases.

If you use KSN with the application, you can enable cloud mode for the application.

Kaspersky Embedded Systems Security switches to using a lightweight version of the malware databases after enabling cloud mode and performing the latest update of the application databases and modules. If cloud mode is disabled, Kaspersky Embedded Systems Security downloads the full version of the application databases from Kaspersky servers during the next update of application databases and modules.

If you are not using KSN or cloud mode is disabled, Kaspersky Embedded Systems Security uses the full version of the application databases.

Cloud mode is disabled automatically if use of KSN is disabled.

Using the KSN Proxy service

User devices managed by Administration Server can communicate with KSN directly or via the KSN Proxy service.

A KSN proxy server provides the following capabilities:

- The user's device can query KSN and submit information to KSN, even without direct access to the Internet.
- A KSN proxy server caches processed data, thereby reducing the load on the external network connection and speeding up receipt of the information that is requested by the user's device.

You can configure the KSN proxy server settings in the Administration Server properties. For details about the KSN proxy server, refer to the Kaspersky Security Center Help.

KSN proxy settings cannot be managed on the command line.

Configuring the use of Kaspersky Security Network in the Web Console

In the Web Console, you can configure the usage of Kaspersky Security Network by the application in the policy properties (**Application settings** → **General Settings** → **Kaspersky Security Network**).

Kaspersky Security Center displays information about the availability of KSN via the client device status (*OK, Critical, Warning*) in the list of managed devices on the **Assets (Devices)** tab.

Kaspersky Security Network settings

Setting	Description
The use of Kaspersky Security Network is enabled / disabled.	By selecting this option, you accept the terms and conditions of the Kaspersky Security Network Statement. You will be able to receive information from Kaspersky's online knowledge base about the reputation of files, web resources, and software. If this check box is cleared, you decline to use Kaspersky Security Network.
Kaspersky Security Network Statement	This link opens the **Kaspersky Security Network Statement** window, where you can read the text of the Kaspersky Security Network Statement.
Use extended KSN mode	If this check box is selected, you will be able to receive information from Kaspersky's online knowledge base about the reputation of files, web resources, and software. Also, anonymous statistics and information about the types and sources of various threats will be sent to Kaspersky to improve Kaspersky Security Network. This check box is unavailable if you are using the *Kaspersky Private Security Network (KPSN)* infrastructure solution.
Use cloud mode	If this check box is selected, Kaspersky Embedded Systems Security uses a lightweight version of the application databases, which reduces the load on operating system resources. Kaspersky Embedded Systems Security downloads the lightweight version of the application databases during the next update following the selection of this check box. If this check box is cleared, Kaspersky Embedded Systems Security uses the full version of the application databases. Kaspersky Embedded Systems Security downloads the full version of the application databases during the next update following the clearing of this check box. The check box is available if use of KSN is enabled. The check box is selected if, when creating a policy, you accepted the terms of the Kaspersky Security Network Statement and are using KSN in extended mode.
Use Administration Server as a KSN proxy server	If this check box is selected, Kaspersky Embedded Systems Security communicates with KSN servers using the KSN Proxy service on the Administration Server. You can manage the settings of the KSN Proxy service in the Administration Server properties.
Use Kaspersky Security Network servers when KSN Proxy is not available	If this check box is selected, Kaspersky Embedded Systems Security communicates with KSN servers directly when the KSN Proxy service is unavailable. This check box is available if the **Use Administration Server as a KSN proxy server** check box is selected.

Configuring the use of Kaspersky Security Network in the Administration Console

In the Administration Console, you can configure the use of Kaspersky Security Network by Kaspersky Embedded Systems Security in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Advanced Threat Protection** → **Kaspersky Security Network**).

You can read the text of the **Kaspersky Security Network Statement** in the Kaspersky Security Network Statement window, which can be opened by clicking the **Kaspersky Security Network Statement** link.

Kaspersky Security Center displays information about the availability of KSN via the client device status (*OK, Critical, Warning*) in the list of managed devices on the **Devices** tab.

Kaspersky Security Network settings

Setting	Description
Kaspersky Security Network Statement	Clicking this link opens the **Kaspersky Security Network Statement** window. In this window, you can read the text of the Kaspersky Security Network Statement.
Kaspersky Security Network (KSN)	This section displays information about the KSN mode or indicates that KSN is not used by Kaspersky Embedded Systems Security. The **Edit** button opens a window where you can configure the use of Kaspersky Security Network.
Enable cloud mode	The check box enables or disables the operating mode in which Kaspersky Embedded Systems Security uses a lightweight version of the malware databases. The check box is available if use of KSN is enabled. The check box is selected if, when creating a policy, you accepted the terms of the Kaspersky Security Network Statement and are using KSN in extended mode. The mode is enabled or disabled after the next application database update.
Use Administration Server as a KSN proxy server	If this check box is selected, Kaspersky Embedded Systems Security communicates with KSN servers using the KSN Proxy service on the Administration Server. This check box is selected by default.
Use KSN servers when KSN Proxy is not available	If this check box is selected, Kaspersky Embedded Systems Security communicates with KSN servers directly when the KSN Proxy service is unavailable. This check box is selected by default. This check box is available if the **Use Administration Server as a KSN proxy server** check box is selected.

Kaspersky Security Network settings

In this window, you can configure Kaspersky Security Network participation settings.

Kaspersky Security Network settings

Setting	Description
More info	Clicking this link opens the Kaspersky website.
Do not use Kaspersky Security Network	By selecting this option, you decline to use Kaspersky Security Network.
Basic KSN mode	By selecting this option, you accept the terms of using Kaspersky Security Network. You will be able to receive information from Kaspersky's online knowledge base about the reputation of files, web resources, and software.

Setting	Description
Extended KSN mode	By selecting this option, you accept the terms of using Kaspersky Security Network. You will be able to receive information from Kaspersky's online knowledge base about the reputation of files, web resources, and software. Also, anonymous statistics and information about the types and sources of various threats will be sent to Kaspersky to improve Kaspersky Security Network.
Kaspersky Security Network Statement	This link opens the **Kaspersky Security Network Statement** window, where you can read the text of the Kaspersky Security Network Statement.

Kaspersky Security Network Statement

In this window, you can read the text of the Kaspersky Security Network Statement and accept its terms and conditions.

Kaspersky Security Network settings

Setting	Description
I confirm that I have fully read, understand, and accept the terms and conditions of the Kaspersky Security Network Statement	By selecting this option, you confirm that you want to use the Kaspersky Security Network, and you have fully read, understood, and accept the terms and conditions of the Kaspersky Security Network Statement that is displayed. This option is available if you selected the **Basic KSN mode** or **Extended KSN mode** option in the **Kaspersky Security Network settings** window.
I do not accept the terms and conditions of the Kaspersky Security Network Statement	By selecting this option, you confirm that you do not want to use Kaspersky Security Network. This option is available if you selected the **Basic KSN mode** or **Extended KSN mode** option in the **Kaspersky Security Network settings** window.

Kaspersky Private Security Network Statement

In this window, you can read the text of the Kaspersky Private Security Network Statement and accept its terms and conditions.

Kaspersky Security Network settings

Setting	Description
I confirm that I have fully read, understand, and accept the terms and conditions of the Kaspersky Security Network Statement	By selecting this option, you confirm that you want to participate in the Kaspersky Security Network, and you have fully read, understood, and accept the terms and conditions of the Kaspersky Private Security Network Statement that is displayed.
I do not accept the terms and conditions of the Kaspersky Security Network Statement	By selecting this option, you confirm that you do not want to use Kaspersky Security Network.

Configuring the use of Kaspersky Security Network on the command line

You can enable or disable the use of Kaspersky Security Network in the command line by using the `UseKSN` option in the general application settings.

You can change the value of `UseKSN` with the help of command line switches or a configuration file that contains all general application settings.

To enable the use of Kaspersky Security Network with the help of command line switches, run:

```
kess-control --set-app-settings UseKSN=<Extended/Basic> --accept-ksn
```

where:

- `<Extended/Basic>`: Kaspersky Security Network mode.
- `--accept-ksn`: an option signifying that you agree to the terms in the Kaspersky Security Network Statement. You confirm that you have fully read, understand, and accept the terms and conditions of the Kaspersky Security Network Statement.

The file ksn_license.<language ID> containing the text of the Kaspersky Security Network Statement is located in the directory /opt/kaspersky/kess/doc/.

To disable the use of Kaspersky Security Network with the help of command line switches, run:

```
kess-control --set-app-settings UseKSN=No
```

To enable or disable use of Kaspersky Security Network with a configuration file, execute the following command:

```
kess-control --set-app-settings --file <path to configuration file> [--json] [--accept-ksn]
```

where:

- `--file <configuration file path>` is the full path to the configuration file with general application settings where the required `UseKSN` value is configured.
- Specify the `--json` option if you are importing settings from a JSON configuration file. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.
- `--accept-ksn`: an option signifying that you agree to the terms in the Kaspersky Security Network Statement. You must specify the option if you are enabling the use of Kaspersky Security Network.

If Kaspersky Embedded Systems Security installed on a client device runs under a Kaspersky Security Center policy, the value of the `UseKSN` setting can only be modified by using Kaspersky Security Center. When Kaspersky Embedded Systems Security installed on a client device stops running under a policy, the following value is assigned to the setting: `UseKSN=No`.

Checking the connection to Kaspersky Security Network using the command line

To check the connection to Kaspersky Security Network, run the following command:

```
kess-control --app-info
```

The **Using Kaspersky Security Network** line displays the status of the connection to Kaspersky Security Network:

- If `Extended KSN mode` is displayed, Kaspersky Embedded Systems Security uses Kaspersky Security Network, information can be obtained from the knowledge base, and anonymous statistics and information about the types and sources of threats are sent.
- If `Basic KSN mode` is displayed, Kaspersky Embedded Systems Security uses Kaspersky Security Network and information can be obtained from the knowledge base, but anonymous statistics and information about the types and sources of threats are not sent.
- If the status is `Disabled`, Kaspersky Embedded Systems Security does not use Kaspersky Security Network.

The **Kaspersky Security Network Infrastructure** line displays information about the infrastructure solution that is used to work with Kaspersky reputation databases: `Kaspersky Security Network` or `Kaspersky Private Security Network`.

A connection to Kaspersky Security Network may be absent for the following reasons:

- The user device is not connected to the internet.
- The use of Kaspersky Security Network is disabled.
- The application has not been activated or the license has expired.
- Problems related to the license key are detected. For example, the key is in the denylist.

Enabling and disabling cloud mode from the command line

Cloud mode is an operating mode of Kaspersky Embedded Systems Security that uses a lightweight version of the malware databases.

You can enable or disable cloud mode in the command line with the help of the `CloudMode=Yes/No` option in the general application settings.

You can edit the value for the setting `CloudMode` using the configuration file that contains all general application settings, or using command line options.

Cloud mode is available if use of Kaspersky Security Network is enabled.

Advanced application settings

You can configure the following additional application settings:

- Using a proxy server in the application.
- Global exclusions to exclude mount points from file operation interception for the File Threat Protection and Anti-Cryptor components and the Malware Scan, Critical Areas Scan, and Removable Drives Scan tasks.
- Exclude process memory from scans.
- File operations interception mode.
- Detection of legitimate applications that threat intruders can use to compromise devices or data.
- Application stability monitoring.
- Application startup settings.
- Limit on the use of memory and processor resources for scan tasks.
- Limit on the use of resident memory by the application.
- Limit on the number of Custom Scan tasks that a non-privileged user can start simultaneously.
- Settings for the transfer of data to the Kaspersky Security Center storage.
- Task management permissions.
- Enabling or disabling namespace scanning.
- Canceling scheduled tasks on a device running on battery power.
- Telemetry additional settings.

Configuring a proxy server

You can configure proxy server settings if the users of the client devices use a proxy server to connect to the internet. The Kaspersky Embedded Systems Security application can use a proxy server to connect to Kaspersky activation servers, to update sources for databases and application modules, to Kaspersky Security Network, and when verifying website certificates using the Web Threat Protection component.

The proxy server is not used by default.

If you use an activation code to activate devices in an isolated network segment without internet access, you can use Kaspersky Security Center Administration Server as a proxy server for access to Kaspersky activation servers.

Configuring proxy server settings in the Web Console

In the Web Console, you can manage proxy settings in the policy properties (**Application settings** → **General Settings** → **Network Settings** and **Application settings** → **General Settings** → **KSC Integration**).

Proxy settings in the Network Settings section

Setting	Description
Proxy server	Proxy server settings for internet access of users of client computers. Kaspersky Embedded Systems Security uses these settings for certain protection components, including the application database and module update.
Use of proxy server enabled / disabled	This toggle switch enables or disables the use of a proxy server by the application.
Proxy server address	Field for entering the proxy server's IP address or domain name.
Proxy server port	Field for entering the proxy server's port. Default value: 3128.
Use proxy server authentication	If this check box is selected, the application uses authentication with a user name and password when connecting to the proxy server. When connecting via an HTTP proxy, we recommend to use a separate account that is not used to sign in to other systems. An HTTP proxy uses an insecure connection, and the account may be compromised.
User name	Entry field for the user name used for proxy server authentication. The entry field is available if the **Use proxy server authentication** check box is selected.
Password	Field for entering the user password for authorization at the proxy server. It is recommended to make sure that the password complexity and anti-bruteforce mechanisms ensure that the password cannot be guessed within 6 months.
Proxy server exclusions	This group of settings contains the **Settings** button, which you can click to open the **Proxy server exclusions** window, in which you can disable the proxy server for connections to the specified addresses. The table contains a list of IP addresses and DNS names for which the proxy server must not be used. You can use masks (`*` and `?` characters) to specify addresses.

Proxy server settings in the KSC Integration section

Setting	Description
Use Kaspersky Security Center as a proxy server for activation	If this check box is selected, the application uses the Kaspersky Security Center Administration Server as a proxy server for connecting to activation servers. This is necessary when activating the application in an isolated network segment without internet access using an activation code. If you are activating the application with a key file, internet access is not necessary.

Configuring proxy server settings in the Administration Console

In the Administration Console, you can manage proxy settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Proxy server settings**).

Proxy server settings

Setting	Description
Do not use proxy server	If this option is selected, the application does not use a proxy server.
Use the specified proxy server settings	If you select this option the application uses the specified proxy server settings to connect to Kaspersky activation servers, to update sources for databases and application modules, to Kaspersky Security Network, and when verifying website certificates using the Web Threat Protection component.
Address and port	Fields for entering the proxy server's IP address or domain name as well as its port. Default port: 3128. These fields are available if the **Use the specified proxy server settings** option is selected.
Use proxy server authentication	This check box enables or disables proxy server authentication using a user name and password. This check box is available if the **Use the specified proxy server settings** option is selected. This check box is cleared by default. When connecting via an HTTP proxy, we recommend to use a separate account that is not used to sign in to other systems. An HTTP proxy uses an insecure connection, and the account may be compromised.
User name	Entry field for the user name used for proxy server authentication. The entry field is available if the **Use proxy server authentication** check box is selected.
Password	Entry field for entering the user password for proxy server authentication. It is recommended to make sure that the password complexity and anti-bruteforce mechanisms ensure that the password cannot be guessed within 6 months. Clicking the **Show** button causes the user's password to be displayed in clear text in the **Password** field. By default, the user password is hidden and is displayed as asterisks. The text box and the button are available if the **Use proxy server authentication** check box is selected.
Use Kaspersky Security Center as a proxy server for application activation	Enables or disables the use of Kaspersky Security Center Administration Server as a proxy server for accessing Kaspersky activation servers. This is necessary when activating the application in an isolated network segment without internet access using an activation code. If this check box is selected, the application gains access to activation servers via the Administration Server that has internet access. This check box is cleared by default.
Proxy server exclusions	This group of settings contains the **Configure** button, which you can click to open the **Proxy server exclusions** window, in which you can specify the list of addresses that can be connected to bypassing the proxy server. As the address, you must specify the IP address or DNS name to which you want to connect without a proxy server. You can use masks (* and ? characters) to specify addresses.

Configuring proxy server settings on the command line

You can enable or disable the use of a proxy server by application components in the command line with the help of the `UseProxy`, `ProxyServer`, and `ProxyBypass` settings in the general application settings.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

The `UseProxy` parameter can take the following values:

- `Yes` — Enable the use of a proxy server.
- `No`: disable the proxy server.

The `ProxyServer` setting lets you specify proxy server settings in the following format: `<connection protocol>://[<user>[:<password>]@]<proxy server address>:<port number>`, where:

- `<user>` is a user name for authentication on the proxy server.
- `<password>` is a user password for authentication on the proxy server.
- `<proxy server address>` is the IP address or domain name of the proxy server.
- `<port number>` is the port number of the proxy server.

Connecting to a proxy server over HTTPS is not supported.

When connecting via an HTTP proxy, we recommend to use a separate account that is not used to sign in to other systems. An HTTP proxy uses an insecure connection, and the account may be compromised.

The `ProxyBypass` setting lets you specify a list of addresses that will be connected to without a proxy server. The address is specified in the `<address>[:<port number>]` format. To specify addresses, you can use masks (* symbols) and comments (after a \ symbol).

Configuring global exclusions

You can configure exclusion of mount points from file operation interception for the File Threat Protection and Anti-Cryptor components, as well as from scanning by the Malware Scan and Critical Areas Scan tasks. Exclusion of mount points allows you to exclude local or remote directories mounted on a device from interception of file operations. In addition, global exclusions affect the Removable Drives Scan task.

Configuring global exclusions in the Web Console

In the Web Console, you can configure use of global exclusions in the policy properties (**Application settings** → **Trusted zone** → **Mount points**).

List of trusted mount points. These are excluded from the interception of file operations by the File Threat Protection and Anti-Cryptor components, as well as from scanning by the Malware scan and Critical areas scan tasks.

Mount points must be specified in the same way as they are displayed in the output of the mount command.

Configuring global exclusions in the Administration Console

In the Administration Console, you can manage global exclusions in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Global exclusions**).

The **Excluded mount points** group of settings contains a **Configure** button. Clicking this button opens the **Excluded mount points** window.

The list in the window contains the paths to the excluded mount points. By default, the list is empty. You can add, edit, and delete items in the list.

Mount point path window

Mount point settings

Setting	Description
File system, access protocol, and path	The settings block lets you set the location of the mount point. In the drop-down list of file systems, you can select the type of file system where the directories that you want to add to scan exclusions are located: • **Local**: local mount points. • **Mounted**: remote directories mounted on the device using the Samba or NFS protocol. • **All remote mounted** – all remote directories mounted on the device using the Samba and NFS protocols.
	If **Mounted** is selected in the drop-down list of file systems, you can select the remote access protocol in the drop-down list on the right: • **NFS**: remote directories mounted on a device using the NFS protocol. • **Samba**: remote directories mounted on a device using the Samba protocol. • **Custom**: all the resources of the device file system specified in the field below.
	If **Local** is selected in the drop-down list of file systems, then in the input field you can enter a path to a mount point that you want to exclude from file operation interception. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name. Mount points must be specified in the same way as they are displayed in the `mount` command output.
Filesystem name	The field for entering the name of the file system where the directories that you to exclude from file operation interception are located. The field is available if the **Mounted** type is selected in the drop-down list of file systems and the **Custom** item is selected in the drop-down list on the right.

Configuring global exclusions on the command line

You can define mount point exclusions in the command line via the `ExcludedMountPoint.item_#` option in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `ExcludedMountPoint.item_#` parameter can take the following values:

- `AllRemoteMounted` — Exclude all remote directories mounted on the device using SMB and NFS protocols from file operation interception.
- `Mounted:NFS` — Exclude all remote directories mounted on the device using the NFS protocol from file operation interception.
- `Mounted:SMB` — Exclude all remote directories mounted on the device using the SMB protocol from file operation interception.
- `Mounted:<file system type>` — Exclude all mounted directories with the specified file system type from file operation interception.
- `/mnt` — Exclude objects in the /mnt mount point (including subdirectories) from file operation interception. This directory is used as the temporary mount point for removable drives.
- `<path that contains the /mnt/user* or /mnt/**/user_share>` mask — Exclude objects in mount points whose names contain the specified **mask** from file operation interception.

  You can use the `*` (asterisk) character to create a file or directory name mask.

  You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`.

  You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`.

  The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask.

  To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk).

  The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself.

  You can use a single `?` character to represent any one character in the file or directory name.

Mount points must be specified in the same way as they are displayed in the `mount` command output.

You can specify several mount points to exclude from scanning.

Excluding process memory from scans

You can configure scan exclusions for process memory. The application does not scan the memory of the specified processes.

Configuring exclusions in the Web Console

In the Web Console, you can configure excluding process memory from scans in the policy properties (**Application settings** → **General Settings** → **Performance**).

Clicking the **Configure exclusion of process memory from scans** link in the **Advanced settings** section opens a window in which you can manage the exclusion list. Specify the path to the processes that the application will exclude from process memory scanning. You can use **masks** to specify the path. By default, the list is empty.

You can use the * (asterisk) character to create a file or directory name mask.

You can indicate a single * character to represent any set of characters (including an empty set) preceding the / character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`.

You can indicate two consecutive * characters to represent any set of characters (including an empty set and the / character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`.

The ** mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask.

To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk).

The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself.

You can use a single ? character to represent any one character in the file or directory name.

Configuring exclusions in the Administration Console

In the Administration Console, you can manage process memory exclusions in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Excluding process memory**).

Clicking **Configure** under **Exclude process memory from scans** opens a window where you can create a list of exclusions. The list in the **Exclude process memory from scans** window contains the paths to processes that the application excludes from process memory scanning. You can use **masks** to specify the path. By default, the list is empty. You can add, edit, and delete items in the list.

You can use the * (asterisk) character to create a file or directory name mask.

You can indicate a single * character to represent any set of characters (including an empty set) preceding the / character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`.

You can indicate two consecutive * characters to represent any set of characters (including an empty set and the / character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`.

The ** mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask.

To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk).

The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself.

You can use a single `?` character to represent any one character in the file or directory name.

You can enable the merging of the list of exclusions when inherited by child policies. To merge the list items, select the **Merge inherited values** check box in the **Exclude process memory from scans** window. If the check box is selected, items in the list of the parent policy are displayed in child policies, and items can be added to the list of the child policy. If the check box is cleared, the list items are not merged when inheriting policy settings. This check box is cleared by default.

Configuring exclusions on the command line

You can configure excluding process memory from scans in the command line using the `MemScanExcludedProgramPath.item_#` option in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

`MemScanExcludedProgramPath.item_#` contains the full path to the process in the local directory. You can use **masks** to specify the path.

You can use the `*` (asterisk) character to create a file or directory name mask.

You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`.

You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`.

The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask.

You can use a single `?` character to represent any one character in the file or directory name.

You can specify several processes to exclude from scanning.

Selecting the interception mode for file operations

The file operation interception mode affects the File Threat Protection and Device Control components.

- For the duration of the scan, the application can block access to files that are being scanned by the File Threat Protection component. By default, access is blocked: any access to the scanned file must wait until the scan results are in. If the scan detects no threats in the file, the application allows access to the file. When detecting infected objects, the application performs the actions specified in the **Action on threat detection** (`ActionOnThreat`) setting of the File Threat Protection component.

  You can choose not to block access to files that are being scanned by the File Threat Protection component. In that case, the scan is performed asynchronously.

- The application can block access to files on the device while the Device Control component is deciding if access to the device can be granted. By default, access is blocked: any access to files on the managed device must wait until the scan results are in. The application allows access to files if after the scan, Device Control allows access to the device that contains the files.

  You can disable file access blocking on the device monitored by the Device Control component. In that case, Device Control determines if access to the device can be allowed in asynchronous mode.

If the blocking mode of file operation interception is not enabled, the File Threat Protection and Device Control components run in notify mode, regardless of the mode that is configured.

Configuring in the Web Console

In the Web Console, you can configure the file operation interception mode in the policy properties (**Application settings** → **General Settings** → **Detection Settings**, **File operation interception mode** section).

The **Block access to files during scans** check box enables or disables the blocking of access to files while they are being scanned by the File Threat Protection and Device Control components.

This check box is selected by default.

If the check box is cleared, access to any file is allowed for the duration of the scan, and the scan runs in asynchronous mode.

Configuring in the Administration Console

In the Administration Console, you can manage application trace settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings** → **File operation interception mode** section).

The **Block access to files during scans** check box enables or disables the blocking of access to files while they are being scanned by the File Threat Protection and Device Control components.

This check box is selected by default.

If the check box is cleared, access to any file is allowed for the duration of the scan, and the scan runs in asynchronous mode.

Configuring on the command line

You can configure the file operation interception mode in the command line using the `FileBlockDuringScan` setting in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `FileBlockDuringScan` parameter can take the following values:

- `Yes` (default value) to block access to files for the duration of the scan by the File Threat Protection and Device Control components.
- `No` to allow access to files during the scan. Requests to any file is allowed, scanning is done asynchronously.

  This file operation interception mode has less impact on the system performance, but there is a risk that a threat in a file will not be disinfected or deleted if the file can, for example, change its name during a scan before the application makes a decision on the status of the file.

Detection of applications that hackers can use to harm

You can enable or disable detection of legitimate applications that intruders can use to compromise devices or data. Such applications do not necessarily have a malicious purpose, but such applications can be exploited by hackers. For detailed information about legitimate applications that intruders can use to compromise the devices or user data, please visit the Kaspersky Encyclopedia website.

Configuring in the Web Console

In the Web Console, you can manage the settings for the detection of legitimate applications that intruders can use to compromise devices or data in the policy properties (**Application settings** → **General Settings** → **Detection Settings** → **Detection of other types of objects**).

The **Legitimate software that can be used by intruders to damage your computer or personal data** check box enables or disables detection of legitimate applications that intruders can use to compromise the device or data of the user. This check box is cleared by default.

In the **Detection Settings** section of the Web Console, you can also configure the file operation interception mode.

Configuring in the Administration Console

In the Administration Console, you can configure detection of legitimate applications that intruders can use to compromise devices or data in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Scan settings** section).

The **Detect legitimate applications that intruders can use to compromise devices or data** check box enables or disables detection of legitimate applications that intruders can use to compromise the device or data of the user. This check box is cleared by default.

Configuring on the command line

In the command line, you can configure the detection of legitimate applications that intruders can use to compromise devices or data by using the `DetectOtherObjects` setting in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `DetectOtherObjects` parameter can take the following values:

- `Yes`: enable detection of legitimate applications that intruders can use to compromise devices or data.
- `No`: do not enable detection of legitimate applications that intruders can use to compromise devices or data.

Enabling application stability monitoring

You can enable or disable the Kaspersky Embedded Systems Security stability monitoring that lets you track the number of times the application terminates abnormally and notify the administrator about the unstable operation of the application, as well as manage monitoring settings.

Configuring in the Web Console

In the Web Console, you can enable or disable application stability monitoring in the policy properties (**Application settings** → **General Settings** → **Performance**, **Advanced settings** section).

The **Enable application stability monitoring** check box enables or disables monitoring of the state of the Kaspersky Embedded Systems Security application.

This check box is cleared by default.

To apply this setting, the application and the Network Agent must be restarted.

If the application is unstable, the following message is displayed in the properties of the device with the installed application: *<Number> abnormal halts of the application since <date and time>*.

Configuring in the Administration Console

In the Administration Console, you can enable or disable application stability monitoring in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Advanced application settings**).

The **Enable application stability monitoring** check box enables or disables monitoring of the state of the Kaspersky Embedded Systems Security application.

This check box is cleared by default.

To apply this setting, the application and the Network Agent must be restarted.

If the application is unstable, the following message is displayed in the properties of the device with the installed application: *<Number> abnormal halts of the application since <date and time>*.

Configuring on the command line

On the command line, you can configure Kaspersky Embedded Systems Security stability monitoring using the `TrackProductCrashes`, `ProductHealthLogFile`, `WarnThreshold`, `WarnAfter_#_crash`, and `WarnRemovingThreshold` settings in the kess.ini configuration file.

The `TrackProductCrashes` allows enabling the sending of device status information from Kaspersky Embedded Systems Security to Kaspersky Security Center when the application is unstable. This setting can take the following values:

- `Yes/true` enables the sending of device status information from Kaspersky Embedded Systems Security to Kaspersky Security Center.
- `No/false` disables the sending of device status information from Kaspersky Embedded Systems Security to Kaspersky Security Center.

The `ProductHealthLogFile` setting lets you specify the path to a file used for application stability monitoring. Default value: /var/opt/kaspersky/kess/private/kess_health.log.

The `WarnThreshold` setting lets you set the time interval (in seconds) in which the application must experience the specified number of abnormal halts before displaying a notification about unstable operation. Default value: 3600 seconds.

The `WarnRemovingThreshold` setting lets you set the time interval (in seconds) after which the application's unstable status will be cleared. Default value: 86400 seconds.

The `WarnAfter_#_crash` setting lets you set the number of abnormal halts of the application after which a notification is displayed about application instability. The setting can take values from 0 to 10. Default value: 10. If the value is 0, an unstable application notification is not displayed.

Configuring application startup settings

You can configure the application startup settings.

Setting a limit in the Web Console

In the Web Console, you can manage application startup settings in the policy properties (**Application settings** → **General Settings** → **User Interaction**, **Application startup settings** section).

Application startup settings

Setting	Description
Maximum consecutive unsuccessful attempts to start the application	The input field for the maximum number of consecutive unsuccessful attempts to start the application. Default value: 5.
Maximum time to wait for application start (min)	The input field for the maximum time to wait for the application to start (in minutes), after which the kess process is restarted. Default value: 3.

In the **User Interaction** section of the Web Console, you can also configure permission to view and manage tasks.

Setting a limit in the Administration Console

In the Administration Console, you can manage application startup settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Application startup settings**).

Under **Application startup settings**, clicking the **Configure** button opens the **Application startup settings** window, in which you can edit the application startup settings (see the table below).

Application startup settings

Setting	Description
Maximum consecutive unsuccessful attempts to start the application	The input field for the maximum number of consecutive unsuccessful attempts to start the application. Default value: 5.
Maximum time to wait for application start (min)	The input field for the maximum time to wait for the application to start (in minutes), after which the kess process is restarted. Default value: 3.

Setting a limit on the command line

On the command line, you can configure application startup settings using the `MaxRestartCount` and `StartupTimeout` settings in the kess.ini configuration file.

The `MaxRestartCount` setting lets you set the maximum number of unsuccessful consecutive attempts to start the application. The setting can take values from 1 to 10. Default value: 5.

The `StartupTimeout` setting lets you set the maximum time to wait for the application to start (in minutes), after which the kess process will be restarted. The setting can take values from 1 to 60. Default value: 3.

Limiting the use of resident memory by the application

You can configure the resident memory usage limit for the application. By default, the limit is set automatically.

Setting a limit in the Web Console

In the Web Console, you can manage the resident memory usage limit in the policy properties (**Application settings** → **General Settings** → **Performance**, **Advanced settings** section).

Under **Advanced settings**, clicking the **Advanced settings** link opens a window in which you can configure the resident memory usage limit (see the table below).

Settings

Setting	Description
Limit application memory usage	This check box limits resident memory usage by the application. This check box is selected by default.
Limit calculation method	In the drop-down list, you can select how to limit resident memory usage: • **Unlimited**. If this option is selected, resident memory usage is not limited. • **Set an exact value (%)**. When this item is selected, the **Exact value (%)** field becomes available, in which you can specify a percentage value. • **Set an exact value (MB)**. When this item is selected, the **Exact value (MB)** field becomes available, in which you can specify a value in megabytes. • **Use the lowest of the two values (% or MB)**. When this item is selected, the **Exact value (%)** and **Exact value (MB)** fields become available, and you can specify values in each. • **Use the highest of the two values (% or MB)**. When this item is selected, the **Exact value (%)** and **Exact value (MB)** fields become available, and you can specify values in each. • **Limit automatically (recommended)**. When this item is selected, resident memory usage is limited automatically (default value).

Setting a limit in the Administration Console

In the Administration Console, you can set a resident memory usage limit in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**).

In the **Advanced application settings** section, clicking the **Configure** button opens the **Additional settings** window, in which you can configure the resident memory usage limit (see the table below).

Settings

Setting	Description
Application memory usage	In the drop-down list, you can select how to limit resident memory usage: • **Unlimited**. If this option is selected, resident memory usage is not limited. • **Limited automatically (recommended)**. When this item is selected, resident memory usage is limited automatically (default value). • **Limited to a percentage of total**. When this item is selected, the **Memory usage limit (%)** field becomes available, in which you can specify the necessary value as a percentage. • **Limited to a value in MB**. When this item is selected, the **Memory usage limit (MB)** field becomes available, in which you can specify the desired value in megabytes. • **Limited to whichever is lowest (%, MB)**. When this item is selected, the **Memory usage limit (%)** and **Memory usage limit (MB)** fields become available, in which you can specify the necessary values. • **Limited to whichever is highest (%, MB)**. When this item is selected, the **Memory usage limit (%)** and **Memory usage limit (MB)** fields become available, in which you can specify the necessary values.
Memory usage limit (%)	Input field for the memory usage limit (as a percentage). Default value: 50.
Memory usage limit (MB)	Input field for the memory usage limit (in megabytes). Default value: 2000.

Setting a limit on the command line

In the command line, you can configure the resident memory usage limit using the `MaxMemory` setting in the kess.ini configuration file.

The `MaxMemory` parameter can take the following values:

- `off` – the resident set size is not limited.
- < `value` >% – a value between 1 and 100 percent of RAM.
- < `value` >`MB` – a value in megabytes.
- `lowest/`< `value` >%/< `value` >`MB` – the smaller of the percentage value and the absolute value in megabytes.
- `highest/`< `value` >%/< `value` >`MB` – the larger of the percentage value and the absolute value in megabytes.
- `auto` – up to 50% of available memory, but not less than 2GB and not more than 16GB.

Default value: `auto`.

Limiting the use of memory and processor resources

You can configure the memory usage limit for scan tasks. The default limit is 8192 megabytes.

> We recommend specifying a memory usage limit for scan tasks below the resident memory usage limit to avoid using up all available memory for scan tasks and causing the application to restart.

You can set a limit on CPU usage for scan tasks. The utilization of device resources during scanning by the Kaspersky Embedded Systems Security application may increase the load on the CPU. This may slow down other applications. To optimize application performance, you can limit the usage of CPU resource. No limit is set by default. When configuring the limit, we recommended keeping in mind that resource of all CPU cores combined is taken as 100%. In the output of the `top` utility, 100% corresponds to the full utilization of a single core. Thus, for a 10-core CPU (reported by the `top` utility as 1000% total), a value of 400% means that the usage of the entire CPU (all cores taken together) is 40%. If you want the value reported by the `top` utility to be kept below 150% on the same 10-core CPU, you need to specify `15` as the CPU usage limit. The CPU usage limit must be an integer from 10 to 100. Moreover, the CPU usage limit is not applied to all application tasks, but only to the *Malware Scan*, *Critical Areas Scan*, and *Inventory* tasks, which means that the CPU usage may exceed the configured value.

Setting a limit in the Web Console

In the Web Console, you can manage the settings for limiting CPU and memory usage for scan tasks in the policy properties (**Application settings** → **General Settings** → **Performance**.

Under **Limit CPU usage by scan tasks** section, you can use the toggle switch to enable or disable the CPU usage limit for the *Malware Scan*, *Critical Areas Scan*, and *Inventory* tasks.

If the check box is selected, the maximum utilization of all processor cores will not exceed the number specified in **Upper limit (%)**. The value in the field must be an integer from 10 to 100.

Under **Advanced settings**, in the **Memory usage limit for scan tasks (MB)** field, you can set a limit on memory usage for scan tasks (in megabytes). Default value: 8192.

Setting a limit in the Administration Console

In the Administration Console, you can enable or disable the CPU usage limit and configure a RAM limit for scan tasks in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Performance** section).

Clicking **Configure** under **Performance** opens the **Performance settings for scan tasks** window, in which you can configure limits in the **CPU and memory usage** section (see table below).

Settings

Setting	Description
Limit CPU usage by scan tasks (%)	The checkbox enables or disables the CPU utilization limit for the *Malware Scan*, *Critical Areas Scan*, and *Inventory* tasks. If the check box is selected, the maximum utilization of all processor cores will not exceed the percentage specified in the field on the right. The value in the field must be an integer from 10 to 100. This check box is cleared by default.
Memory usage limit for scan tasks (MB)	Input field for the memory usage limit for scan tasks (in megabytes). Default value: 8192.

Setting a limit on the command line

On the command line, you can configure CPU utilization limits for tasks of the *ODS* and *InventoryScan* types using the `UseOnDemandCPULimit` and `OnDemandCPULimit` settings in the general application settings.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

The `UseOnDemandCPULimit` parameter can take the following values:

- `Yes`: enable the CPU usage limit for tasks of the *ODS* and *InventoryScan* types.
- `No`: disable the CPU usage limit for tasks.

The `OnDemandCPULimit` option sets the maximum utilization level for all processor cores (as a percentage) when running *ODS* and *InventoryScan* tasks. The value in the field must be an integer from 10 to 100. Default value: 100.

On the command line, you can configure memory usage limits for certain task types (*ODS* and *InventoryScan*) using the `ScanMemoryLimit` setting in the kess.ini configuration file. Default value: 8192.

Limiting the number of Custom Scan tasks

You can set a limit on the number of custom scan tasks that a non-privileged user can simultaneously run on a device. There is no limit on the number of tasks that a user with root privileges can run.

Setting a limit in the Administration Console

In the Administration Console, you can configure the limit for the number of concurrent custom scan tasks that a non-privileged user can run on the device in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Performance** section).

Clicking the **Configure** button in the **Performance** section opens the **Performance settings for scan tasks** window, in which you can specify a value from 0 to 100,000 in the **Number of custom scan tasks** section, **Maximum number of custom scan tasks** field. Default value: 0. If 0 is specified, a non-privileged user cannot start custom scan tasks.

Setting a limit on the command line

You can configure the limit on the number of concurrent custom scan tasks on the command line using the `LimitNumberOfScanFileTasks` setting in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

`LimitNumberOfScanFileTasks` accepts values from 0 to 100000. If 0 is specified, a non-privileged user cannot start custom scan tasks. Default value: 5.

Configuring the transfer of data to Kaspersky Security Center storage

In Kaspersky Security Center, you can enable or disable the transfer of data about files in Backup, quarantined files, unprocessed files, and connected devices to the Kaspersky Security Center storage.

The general list of files quarantined by Kaspersky applications on client devices is kept in Kaspersky Security Center and is available in the Web Console (**Operations** → **Repositories** → **Quarantine**) and the Administration Console (**Advanced** → **Repositories** → **Quarantine**).

A general list of files placed in Backup by Kaspersky applications on client devices is kept in Kaspersky Security Center and is available in the Web Console (**Advanced** → **Repositories** → **Backup**) and in the Web Console (**Operations** → **Repositories** → **Backup**).

Information about unprocessed files is displayed in the list of active threats in the Web Console (**Operations** → **Repositories** → **Active threats**) and in the Administration Console (**Advanced** → **Repositories** → **Active threats**).

Information about devices installed on or connected to a client device is displayed in the list of hardware in the Web Console (**Operations** → **Repositories** → **Hardware**) and in the Administration Console (**Advanced** → **Repositories** → **Hardware**). Data is transferred if Device Control is enabled.

Information about applications discovered on the client device is displayed in the list of applications in the Web Console (**Operations** → **Third-party applications** → **Applications registry**) and in the Administration Console (**Advanced** → **Application management** → **Applications registry**). Data is transferred if Application Control is enabled.

Enabling or disabling the transfer of data in the Web Console

In the Web Console, you can manage Administration Server integration settings in the policy properties (**Application settings** → **General Settings** → **KSC Integration**, **Data transfer to Administration Server** section, **Advanced settings** link) and in the properties of the managed device (**Assets (Devices)** → **Managed devices** → <device name> link → **Applications** tab → <Kaspersky Embedded Systems Security application name> link → **Application settings** tab → **Storage settings** section, **Settings for transferring data to the Administration Server** section).

Settings for transferring data to the Administration Server

Setting	Description
Transfer data to the Administration Server	Event categories on client devices, information about which must be sent to the Administration Server. You can disable the transfer of some data to optimize disk space on the Administration Server. The application sends events of the following categories: • **About files in Quarantine** • **About files in Backup**. • **About unprocessed files**. • **About installed devices**. • **About started applications**.

Enabling and disabling the transfer of data in the Administration Console

In the Administration Console, you can enable or disable data transfer to the Administration Server in policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Storage settings**, **Transfer data to the Administration Server**) or in the properties of the managed device (**Managed devices** → **Devices** tab → device properties window → **Applications** section → application properties window → **Storage settings**, **Transfer data to the Administration Server** section).

Clicking **Configure** in the **Transfer data to the Administration Server** section opens the **Data transfer settings**, in which you can enable or disable the transfer of data to the Administration Server.

Settings for transferring data to the Administration Server

Setting	Description
About files in Quarantine	This check box enables or disables the transfer of data about quarantined files to the Administration Server. This check box is selected by default.
About files in Backup	This check box enables or disables the transfer of data about files in Backup to the Administration Server. This check box is selected by default.

Setting	Description
About unprocessed files	This check box enables or disables sending notifications about the files, which were not processed during the scan, to the Administration Server. This check box is selected by default.
About installed devices	This check box enables or disables the transfer of data about devices installed on a client device or connected to it, to the Administration Server. This check box is selected by default.
About applications found on the device	This check box enables or disables the sending of the list of applications that the Inventory task discovers on the client device to the Administration Server. This check box is selected by default.

Configuring permissions for task management

You can define the following user permissions in Kaspersky Security Center:

- Viewing local tasks created in Kaspersky Embedded Systems Security If the Kaspersky Security Center policy prohibits users from viewing and editing local tasks, information about the Scan_My_Computer, Critical_Areas_Scan, Inventory_Scan, Update, and Rollback tasks is not available.
- Viewing tasks, created in Kaspersky Security Center, on client devices

Configuring in the Web Console

In the Web Console, you can manage the permissions to view and manage tasks in the policy properties (**Application settings** → **General Settings** → **User Interaction**, **Permitted actions** section).

Task management settings

Setting	Description
Access to local tasks	This check box allows or blocks the users from viewing local tasks created in Kaspersky Embedded Systems Security and control of these tasks on the managed client devices. If this check box is selected, users can configure and run tasks in the graphical interface of the application and on the command line. If this check box is cleared, local tasks are not used. Scheduled local tasks do not run. Tasks cannot be run or configured in the graphical interface of the application or on the command line. This check box is cleared by default.
Manage group tasks	This check box allows or prohibits the users from viewing the tasks created in Kaspersky Security Center and managing these tasks on client devices. If this check box is selected, tasks created in Kaspersky Security Center are displayed in the graphical interface of the application. Users can start and stop these tasks from the graphical interface of the application and from the command line. If this check box is cleared, tasks created in Kaspersky Security Center cannot be viewed or managed in the graphical interface of the application or on the command line. This check box is cleared by default.

In the **User Interaction** section of the Web Console, you can also configure the application startup settings.

Configuring in the Administration Console

In the Administration Console, you can manage permissions to view tasks in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Local Tasks → Task management**).

Task management settings

Setting	Description
Allow local tasks	This check box allows or blocks the users from viewing local tasks created in Kaspersky Embedded Systems Security and control of these tasks on the managed client devices. If this check box is selected, users can configure and run tasks in the graphical interface of the application and on the command line. If this check box is cleared, local tasks are not used. Scheduled local tasks do not run. Tasks cannot be run or configured in the graphical interface of the application or on the command line. This check box is cleared by default.
Allow viewing and managing tasks created through KSC	This check box allows or prohibits the users from viewing the tasks created in Kaspersky Security Center and managing these tasks on client devices. If this check box is selected, tasks created in Kaspersky Security Center are displayed in the graphical interface of the application. Users can start and stop these tasks from the graphical interface of the application and from the command line. If this check box is cleared, tasks created in Kaspersky Security Center cannot be viewed or managed in the graphical interface of the application or on the command line. This check box is cleared by default.

Enabling or disabling monitoring of namespaces

You can enable or disable the use of the system namespace mechanism. This mechanism is used in container systems and sessions with mandatory access control in Astra Linux operating systems.

Configuring namespace monitoring in the Web Console

In the Web Console, you can enable or disable the namespace mechanism in the policy properties (**Application settings → General Settings → Container Scan settings**).

The **Namespace scan enabled / disabled** toggle switch enables or disables the use of the namespace mechanism on supported operating systems.

The toggle switch is switched on by default.

Configuring namespace monitoring in the Administration Console

In the Administration Console, you can enable or disable the namespace mechanism in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings → Container Scan settings**).

The **Enable monitoring of namespaces and containers** check box enables or disables the use of the namespace mechanism on supported operating systems.

This check box is selected by default.

Configuring namespace monitoring on the command line

You can enable or disable the use of namespace mechanism on the command line by using the `NamespaceMonitoring` setting in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `NamespaceMonitoring` parameter can take the following values:

- `Yes` (default value) – Enable the namespace mechanism.
- `No` – disable the namespace mechanism.

Canceling tasks when running on battery power

You can enable or disable the canceling of scheduled tasks when running on battery power. Canceling scheduled tasks can reduce resource consumption and extend battery life.

If cancellation of scheduled tasks is enabled and the device is running on battery power, Kaspersky Embedded Systems Security cancels the following scheduled tasks:

- All scan tasks (custom scan, critical areas scan, malware scan)
- Inventory
- Databases and application modules update
- System Integrity Check

You can run tasks manually as needed.

Configuring in the Web Console

In the Web Console, you can configure the cancellation of scheduled tasks in the policy properties (**Application settings** → **General Settings** → **Performance**).

The **Postpone or skip scheduled tasks while running on battery power** check box enables or disables the cancellation of scheduled tasks.

When enabled, the application enters the power saving mode when running on battery power and cancels scheduled tasks.

If this check box is cleared, all tasks run as scheduled, regardless of whether the device is running on battery power.

This check box is selected by default.

Configuring in the Administration Console

In the Administration Console, you can manage the cancellation of scheduled tasks in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**, **Performance**).

The **Skip scheduled tasks when running on battery power** check box enables or disables the cancellation of scheduled tasks.

This check box is selected by default.

If this check box is cleared, all tasks run as scheduled, regardless of whether the device is running on battery power.

Configuring on the command line

You can configure the cancellation of scheduled tasks on the command line using the `PowerSaveMode` setting in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `PowerSaveMode` parameter can take the following values:

- `Yes` (default value) skips scheduled tasks when the device is running on battery power.
- `No` runs scheduled tasks regardless of whether the device is running on battery power or not.

Managing additional telemetry settings

You can manage the following advanced settings for the obtaining of telemetry sent to the Kaspersky Managed Detection and Response solution when integrated with it:

- Telemetry source. You can use the eBPF technology and the auditd service to get telemetry, or the eBPF technology alone.
- The mode of the auditd service. Kaspersky Embedded Systems Security can use the auditd service in exclusive mode or in non-exclusive mode.

Configuring in the Web Console

In the Web Console, you can manage additional telemetry settings in the policy properties (**Application settings** → **General Settings** → **OS interaction settings**, **Telemetry settings**).

Additional telemetry settings

Setting	Description
Telemetry source	The source that Kaspersky Embedded Systems Security uses to obtain telemetry: • **Use only eBPF**. Only eBPF technology is used. • **By default**. Both eBPF technology and the auditd service are used.
Operating mode for auditd	The mode of the auditd service: • **Exclusive**. If this option is selected, Kaspersky Embedded Systems Security uses the auditd service in exclusive mode. • **Non-exclusive**. If this option is selected, the auditd service is used in non-exclusive mode. This setting is applied only if **Telemetry source** is set to **By default**.

Configuring in the Administration Console

In the Administration Console, you can manage additional telemetry settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** →**OS interaction settings**, **Telemetry additional settings**).

Additional telemetry settings

Setting	Description
Telemetry source	The source that Kaspersky Embedded Systems Security uses to obtain telemetry: • **Use only eBPF** means only the eBPF technology is used for obtaining telemetry. • **Use eBPF and auditd** (default) means the eBPF technology and auditd service are used to get telemetry.
Using auditd	The mode of the auditd service: • Use AuditD **in exclusive mode** (default) means Kaspersky Embedded Systems Security uses the auditd service in exclusive mode. • **Non-exclusive mode** means the auditd service is used in non-exclusive mode. This setting is applied only if **Telemetry source** is set to **Use eBPF and auditd**.

Configuring on the command line

On the command line, you can manage additional telemetry settings using the `TelemetrySource` and `UseTelemetryMonopolyMode` settings of the Behavior_Detection predefined task.

You can change the settings using the `kess-control --set-settings 20` command, by editing the configuration file that contains all the settings of the task, or using the command line options in the `< setting name >=< setting value >` format.

The `TelemetrySource` setting lets you specify the telemetry source. This setting can take the following values:

- `Default` (default) means the eBPF technology and auditd service are used to get telemetry.
- `OnlyEBPF` means that only the eBPF technology is used to get telemetry.

The `UseTelemetryMonopolyMode` setting lets you specify the usage mode of the auditd service if it is used for getting telemetry. This setting can take the following values:

- `True` (default) means Kaspersky Embedded Systems Security uses the auditd service in exclusive mode.
- `False` means the auditd service is used in non-exclusive mode.

> The `UseTelemetryMonopolyMode` setting is applied only if `TelemetrySource` is set to `Default`.

Trusted zone

A trusted zone is a list of objects and applications that the application does not scan or monitor; this list is created by the administrator.

The system administrator decides what to include in the trusted zone depending on which objects need to be worked with as well as which applications are installed on the device. It may become necessary to include objects and applications in the trusted zone if, for example, the application blocks access to an object or trusted application, but you are confident that the object or application is harmless.

You can add the following objects to the trusted zone:

- File Threat Protection exclusions
- Trusted web addresses for Web Threat Protection
- Trusted IP addresses for Network Threat Protection
- Network ports that the application will monitor
- Trusted root certificates
- Trusted domains that are excluded from scanning by the application
- Trusted mount points (global exclusions), that is, mount points excluded from file operation interception for the File Threat Protection and Anti-Cryptor components and the Malware Scan, Critical Areas Scan, and Removable Drives Scan tasks
- Traffic interception exclusions
- Processes excluded from Behavior Detection
- Trusted devices, that is, devices to which full access is allowed at any time for specified users

Backup

If Kaspersky Embedded Systems Security detects malicious code in a file while scanning a protected device, the application can block the file, assign the *Infected* status to it, place a copy in Backup, and attempt to disinfect the file.

Backup keeps copies of files that have been deleted or modified during disinfection. A *backup copy* is created before disinfecting or deleting the file. Backup copies of files are stored in a special format and do not pose a threat.

By default, the /var/opt/kaspersky/kess/common/objects-backup/ directory is used to store backup copies of files. This directory also stores quarantined files. You can change the directory for storing Backup and Quarantine objects using the command line.

Sometimes it is not possible to maintain the integrity of files during disinfection. If after disinfection, you partially or completely lose access to important information in a disinfected file, you can try restoring the file from the backup copy.

Restoring infected objects may lead to a device infection.

Backup file copies may contain personal data. Root privileges are required to access Backup objects.

You can manage Backup settings on a device using a policy in the Web Console, in the Administration Console or using the command line. You can configure the following Backup settings:

- Storage time for objects in Backup. Objects are kept for 30 days by default.
- Maximum Backup size. Backup has an unlimited size by default.

Path to the directory where Backup and Quarantine objects are stored (can only be configured on the command line).

When the specified retention period expires, or when the maximum Backup size is reached, the application automatically deletes the oldest backup copies regardless of their status.

You can delete the backup copy of a restored or unrestored file manually.

A general list of files placed in Backup by Kaspersky applications on client devices is generated in Kaspersky Security Center and is available in the Administration Console (**Advanced** → **Repositories** → **Backup**) and in the Web Console (**Operations** → **Repositories** → **Backup**). You can view the properties of files in backup storage on protected devices and delete files from it. Kaspersky Security Center does not copy files from Backup to the Administration Server; all files are stored in the Backups on protected devices. File restore takes place on the protected device.

The file is restored from Backup to its original location in accordance with to the specified settings. Restoration preserves the MAC label (only on Astra Linux devices) and attributes of the original file (other than the immutable and append-only attributes).

On Astra Linux devices, restoring a file from Backup ends with an error in the following cases:

- The MAC label of the file is higher than the MAC label of the directory to which the file is being restored.
- The MAC label of the file is lower than the MAC label of the directory to which the file is being restored, and the ccnr flag is not set for the directory.

To manage files in Backup in Kaspersky Security Center, you need to enable the transfer of data about files in Backup to the Administration Server.

Configuring Backup settings in the Web Console

In the Web Console, you can manage Backup settings in the policy properties (**Application settings** → **General Settings** → **Reports and Storage**) and in managed device properties (**Assets (Devices)** → **Managed Devices** → <device name> link → **Applications** tab → <Kaspersky Embedded Systems Security application name> link → **Application settings** tab → **Storage settings** section).

Backup settings

Setting	Description
Delete objects after (days)	If the check box is selected, the maximum file storage term is limited to the defined time interval. When the maximum retention period expires, the application deletes the oldest files from Backup. If the check box is cleared, the period for storing objects in Backup is unlimited. Default value: 30.
Limit the size of Backup to (MB)	If the check box is selected, the maximum storage size is limited to the defined value. When the size of files in Backup reaches the maximum size, the application automatically deletes the oldest files to keep the size of files below the maximum Backup size. If the check box is cleared, the size of Backup is unlimited. Possible values: 1–4000. Default value: 1024.

Configuring Backup settings in the Administration Console

In the Administration Console, you can manage Backup settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Storage settings**) and in managed device properties (**Managed devices** → **Devices** tab → device properties window → **Applications** section → application properties window → **Storage settings**).

Backup settings

Setting	Description
Store objects no longer than (days)	This check box enables or disables the storage period limit (in days) for the objects in the Backup storage. Possible values: 0–10000. If 0 is specified, the period for storing objects in the Backup storage is unlimited. Default value: 30.
Limit the size of Backup to (MB)	This check box enables or disables the maximum Backup storage size (in megabytes). Possible values: 1–4000. If the check box is cleared, the size of Backup is unlimited. This check box is cleared by default.

Configuring Backup settings on the command line

On the command line, you can configure Backup settings using the `BackupDaysToLive, BackupSizeLimit` and `QuarantineBackupFolder` settings in the general application settings.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

Backup settings

Setting	Description	Value
`BackupDaysToLive`	Time period for storing objects in the Backup storage (in days). After the specified time has elapsed, the application deletes the oldest backup copies of files. To remove the object retention limit, set 0.	0 to 10000 0–unlimited retention. Default value: 30.
`BackupSizeLimit`	Maximum Backup size in MB. When the maximum Backup storage size is reached, the application deletes the oldest backup copies of files. To remove the Backup size limit, set 0.	0 to 4000 0–unlimited size. Default value: 0.
`QuarantineBackupFolder`	Path to the directory in which the Backup and Quarantine objects are stored. You can specify a directory that is different from the default directory. You can use a directory on any device as the directory for storing Backup and Quarantine objects. It is not recommended to assign directories that are located on remote devices, such as those mounted via the Samba and NFS protocols. If the specified directory does not exist or is unavailable, the application uses the default directory.	Default value: /var/opt/kaspersky/kess/common/objects-backup/ Root privileges are required for access to the default directory.

Working with Backup objects on the command line

You can use the Backup management commands in the command line for the following actions on Backup objects:

- View Backup object details.
- Delete some or all objects from Backup.
- Restore objects from Backup.

Restoring infected objects may lead to a device infection.

Viewing Backup object details

To view the details of the objects in Backup, run:

```
kess-control -B --query ["<filter conditions>"] [-n <number>] [--json]
```

where:

- `< filter conditions >` specifies one or more logical expressions in the `< field > < comparison operation > '< value >'` format combined using the `and` logical operator to restrict query results. If you do not specify any filter conditions, the application displays the details of all objects in Backup.
- `< number >` specifies how many of the latest objects in storage you want to display. If you do not specify the `-n` option, the application displays information about the last 30 objects. If you want to display information about all objects, specify `0`.
- `--json`: output data in JSON format.

The `ObjectId` line displays the numeric identifier that the application assigned to the object when placing it in Backup. This ID is used to perform actions on the object, such as restoring or removing the object from the Backup storage.

Restoring objects from Backup

To restore an object under its original name to its original location, execute the following command:

```
kess-control -B --restore < object ID >
```

where `< object ID >` is the numeric identifier that the application assigned to the object when placing it in Backup.

To restore an object under a new name to a specified directory, execute the following command:

```
kess-control -B --restore < object ID > --file < file path >
```

where `--file < file path >` is the new name of the file and the path to the directory you want to save it to. If the specified directory does not exist, the application creates it.

Deleting objects from Backup

To remove selected objects from Backup, run:

```
kess-control -B --mass-remove --query "< filter conditions >"
```

where `< filter conditions >` is one or several logical expressions in the format `< field > < comparison operator > '< value >'`, combined with the help of the logical operator `and` to limit the results.

Examples:

To remove an object with ID=15:

```
kess-control -B --mass-remove --query "ObjectId == '15'"
```

To remove objects that contain "test" in their names or paths:

```
kess-control -B --mass-remove --query "FileName like '%test%'"
```

To remove all objects from Backup, run:

```
kess-control -B --mass-remove
```

Integration with Kaspersky Managed Detection and Response

Kaspersky Managed Detection and Response is a solution for automatically detecting and analyzing security incidents in your infrastructure using telemetry and advanced machine learning technologies. Information about the incident is sent to Kaspersky experts, who can then either process the incident themselves or provide recommendations on how to resolve it.

When interacting with Kaspersky Managed Detection and Response, Kaspersky Embedded Systems Security can:

- Send telemetry data to Kaspersky Managed Detection and Response for threat detection.
- Perform threat response actions to maintain security when receiving commands from Kaspersky Managed Detection and Response.

Integration of Kaspersky Embedded Systems Security with the Kaspersky Managed Detection and Response solution is facilitated by the Managed Detection and Response (MDR) component of the application.

To use the Kaspersky Managed Detection and Response functionality, you need to activate the MDR component by adding an MDR license key using a key file or activation code.

> The license that allows to activate the Kaspersky Managed Detection and Response functionality must first be registered on the MDR portal.

If your configuration does not satisfy the requirements for operation without KPSN configuration files, then to integrate with Kaspersky Managed Detection and Response, you need to upload a KPSN configuration file to the Kaspersky Security Center Administration Server. This file contains the necessary telemetry settings. The KPSN configuration file with the .pkcs7 extension is located in the ZIP archive of the MDR configuration file. For details, refer to Kaspersky Managed Detection and Response Help.

Integration with Kaspersky Managed Detection and Response involves the following steps:

1. **Enabling required components of Kaspersky Embedded Systems Security**

   Make sure that the following components of Kaspersky Embedded Systems Security are enabled and working:

   - File Threat Protection.
   - Behavior Detection.

   If these components are disabled, the device will have a red status in Kaspersky Managed Detection and Response.

   We also recommend enabling Web Threat Protection and Network Threat Protection. If these components are disabled, the device will have a yellow status in Kaspersky Managed Detection and Response.

   See the Kaspersky Managed Detection and Response Help for more information about device statuses.

2. **Enabling the use of Kaspersky Security Network in extended mode.**

   The Kaspersky Managed Detection and Response functionality is not available if Kaspersky Security Network is disabled or being used in standard mode. You can configure the use of Kaspersky Security Network in the Web Console, the Administration Console, or on the command line.

3. **Activating the MDR component**

   Make sure to add the MDR license key to the application.

4. **Uploading the KPSN configuration file**

   If your configuration does not satisfy the requirements for operation without KPSN configuration files, you need to upload a KPSN configuration file to the Kaspersky Security Center Administration Server.

5. **Enabling the MDR component**

   The MDR component is disabled by default. You can enable or disable the component:

   - Using the Web Console.
   - Using the Administration Console.
   - Using the command line.

   You can check the status of the MDR component:

   - In the device properties in the Web Console or the Administration Console.
   - Using the command line.

About response actions

When integrated with Detection and Response solutions, Kaspersky Embedded Systems Security can perform threat response actions to maintain security functionality.

Kaspersky Embedded Systems Security can perform the following threat response actions:

- Isolate a device from the network.

  Network isolation can be enabled or disabled on the side of the Kaspersky Managed Detection and Response solution. You can manually disable the network isolation of a device on the command line.

- Quarantine file.

  The action is performed using the *Quarantine file* task. The task is created and run on the side of the Kaspersky Managed Detection and Response solution. You can also manually quarantine files on the command line.

- Restore a file from Quarantine.

  The response action is generated on the side of the Kaspersky Managed Detection and Response solution. You can also manually restore a file from quarantine using Kaspersky Security Center or the command line.

- Receive file from device.

  This action is performed using the *Get file* task. For example, you can configure the application to get an event log file generated by a third-party application. The task is created and run on the side of the Kaspersky Managed Detection and Response solution.

- Remotely run a process on a device.

  This action is performed using the *Run application* task. For example, you can remotely run a utility that creates a device configuration file, and then retrieve the created file using the *Get file* task. The task is created and run on the side of the Kaspersky Managed Detection and Response solution.

- Remotely terminate a process on a device.

  The action is performed using the *Kill process* task. For example, you can remotely terminate an Internet speed test utility that was launched using the "Run process" task. The task is created and run on the side of the Kaspersky Managed Detection and Response solution.

Uploading the KPSN configuration file for integration with Kaspersky Managed Detection and Response

If your configuration does not satisfy the requirements for operation without KPSN configuration files, then to integrate with Kaspersky Managed Detection and Response, you need to upload a *KPSN configuration file* to the Kaspersky Security Center Administration Server. This file contains the necessary telemetry settings. The KPSN configuration file with the .pkcs7 extension is located in the ZIP archive of the MDR configuration file. For details, refer to Kaspersky Managed Detection and Response Help.

You can upload the KPSN configuration file in the Web Console or the Administration Console.

By uploading the KPSN configuration file, you agree that data from the device where Kaspersky Embedded Systems Security is installed will be automatically sent to Kaspersky for processing. Do not load the configuration file if you do not agree that the transmitted data will be processed. For detailed description of the transmitted data, refer to Kaspersky Managed Detection and Response documentation.

To upload the KPSN configuration file using the Web Console:

1. In the main Web Console window, open the Administration Server properties window.
2. In the list on the left, select the **KSN proxy server settings** section.
3. Turn on the **Enable KSN proxy server on the Administration Server as a proxy server** toggle switch to enable the KSN proxy server service.
4. Turn on the **Use Kaspersky Private Security Network** toggle switch.
5. In the window that opens and displays a warning about the specific aspects of using the KSN proxy server on the distribution points with the previous version of the Network Agent installed, click **OK**.
6. Click the **File with KSN proxy server settings** button.
7. Select the KSN configuration file (with the pkcs7 extension) and click **Open**.
8. Accept the terms and conditions of the KPSN Statement and click the **Save** button.

To upload the KPSN configuration file using the Administration Console:

1. In the Administration Console tree, open the Administration Server properties window.
2. Select **KSN proxy server → KSN proxy server settings**.
3. Select the **Use Administration Server as a proxy server** check box to enable the KSN proxy server service.
4. Select the **Configure KPSN** check box.

5. In the window that opens and displays a warning about the specific aspects of using the KSN proxy server on the distribution points with the previous version of the Network Agent installed, click **OK**.

6. Click the **File with KSN proxy server settings** button.

7. Select the KSN configuration file (with the pkcs7 extension) and click **Open**.

8. Accept the terms and conditions of the KPSN Statement and click the **Apply** button.

Configuring the Kaspersky Managed Detection and Response integration in the Web Console

In the Web Console, you can manage Kaspersky Managed Detection and Response integration settings in the policy properties (**Application settings** → **Built-in Agents Configuration** → **Managed Detection and Response**).

To integrate with Kaspersky Managed Detection and Response, the File Threat Protection and Behavior Detection protection components must be enabled. We also recommend enabling Web Threat Protection and Network Threat Protection.

An MDR license key is required to activate the MDR component.

MDR integration settings

Setting	Description
Managed Detection and Response enabled / disabled	This toggle switch enables or disables the MDR component, which is necessary for integrating Kaspersky Embedded Systems Security with the Kaspersky Managed Detection and Response solution. The component is disabled by default. To activate the MDR component, you need to enable the Extended mode of Kaspersky Security Network.

Configuring the Kaspersky Managed Detection and Response integration in the Administration Console

In the Administration Console, you can enable or disable the MDR component in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Detection and Response** → **Managed Detection and Response**).

An MDR license key is required to activate the MDR component.

MDR integration settings

Setting	Description
Enable Managed Detection and Response	This check box enables or disables the MDR component, which is necessary for integrating Kaspersky Embedded Systems Security with the Kaspersky Managed Detection and Response solution. The component is disabled by default. If you enable the MDR component and Kaspersky Security Network is disabled or used in standard mode, you are prompted to accept the Kaspersky Security Network Statement to enable Kaspersky Security Network in Extended mode. If changing KSN settings is prohibited by parent policy, an error message is displayed and the MDR component is not enabled.

Configuring the Kaspersky Managed Detection and Response integration on the command line

In the command line, you can do the following:

- Enable or disable the MDR component.
- Add or remove the MDR component activation license key (MDR key) using the commands for managing license keys.
- Change the start time of the *Mdr_Autostart_Scan* service task, which is created automatically after the integration of Kaspersky Embedded Systems Security with Managed Detection and Response. When created, the *Mdr_Autostart_Scan* task is set to run once a day. You can set the start time if needed. No other task settings or schedule options can be edited.

We recommend configuring the integration between Kaspersky Embedded Systems Security and Kaspersky Managed Detection and Response in the Administration Console or in the Web Console.

You can enable or disable the MDR component by editing the `UseMDR` option in the general application settings. You can edit the option via command line switches or a configuration file that contains all general application settings.

The `UseMDR` parameter can take the following values:

- `Yes` enables the MDR component.
- `No` disables the MDR component.

If you enable the MDR component, and Kaspersky Security Network is disabled or used in Basic mode, an error message is displayed and the MDR component is not enabled. To enable the MDR component, you must enable KSN in Extended mode.

Network isolation of a device

When integrated with the Kaspersky Managed Detection and Response solution, a device can be isolated from the network as part of a threat response action.

Special considerations involved with network isolation

After enabling network isolation, the application severs all active network connections on the device and blocks all new TCP/IP network connections, except for the connections listed below:

- Connections specified in exclusions from network isolation
- Connections initiated by Kaspersky Embedded Systems Security services
- Connections initiated by the Kaspersky Security Center Network Agent
- Connections with KSN

Network isolation can only be applied if the integration of Kaspersky Embedded Systems Security with the Kaspersky Managed Detection and Response solution is enabled and the MDR component is activated.

Enabling and disabling network isolation of a device, as well as configuring exclusions from network isolation, is performed on the side of the Kaspersky Managed Detection and Response solution. For details, refer to Kaspersky Managed Detection and Response Help.

If necessary, you can manually disable the network isolation of a device on the command line. Manually disabling network isolation is possible regardless of whether integration with the Kaspersky Managed Detection and Response solution is enabled, and regardless of whether a policy is applied to the device. You can check the network isolation status of a device on the command line.

Network isolation limitations

When you use network isolation, we strongly recommended that you familiarize yourself with the relevant limitations.

For network isolation to work, Kaspersky Embedded Systems Security must be running. If Kaspersky Embedded Systems Security malfunctions (and the application is not running), traffic may not be blocked when network isolation is enabled by Kaspersky Managed Detection and Response.

DHCP and DNS are not automatically added to network isolation exclusions, so if the network address of a resource changes during network isolation, Kaspersky Embedded Systems Security cannot gain access to it.

The proxy server is not added automatically to the network isolation exclusions, so you need to add it to the exclusions manually so that Kaspersky Embedded Systems Security does not lose contact with the server providing the integration.

Excluding a process from network isolation by name is supported on devices with kernel version 4.18 or later with eBPF and BTF support and with tracepoints and kprobes functions enabled for kernel tracing.

When using network isolation, it is recommended to:

- Use a KSN proxy server to interact with Kaspersky Security Network.
- Use Kaspersky Security Center as a proxy server for application activation.

  If it is impossible to use Kaspersky Security Center as a proxy server, you must configure the proxy server that you want to use and add it to exclusions.
- Specify Kaspersky Security Center as the database update source.

Managing the Quarantine

Quarantine is a special storage location on the device for files that may be infected with viruses or cannot be disinfected at the time of detection. Quarantine allows isolating a file for further investigation. Quarantined files are stored in an encrypted form and do not threaten the security of the device. In contrast to Quarantine, Backup stores backup copies of files, in which malicious code was detected and which were deleted or modified during the disinfection process.

The application only uses Quarantine when integrated with the Kaspersky Managed Detection and Response solution to perform recommended threat response actions. You can also manually quarantine files that you consider dangerous to your device.

By default, the /var/opt/kaspersky/kess/common/objects-backup/ directory is used to store quarantined files. This directory also contains Backup objects. You can change the directory for storing Backup and Quarantine objects using the command line.

Quarantined files may contain personal data. Root privileges are required to access quarantined files.

You can configure quarantine settings on a device using a policy in the Web Console or in the Administration Console or using the command line. You can configure the following Quarantine settings:

- The percentage of Quarantine that must be full to generate an event about Quarantine being full. By default, an event is generated when the Quarantine is 90% full.
- Maximum size of the Quarantine in megabytes. When the maximum Quarantine size is reached, the application automatically deletes the oldest quarantined files. By default, the maximum quarantine size is 200 MB.
- Path to the directory where Backup and Quarantine objects are stored (can only be configured on the command line).

Quarantining files

Some files can be critically important for the operation of the operating system and the application. Quarantining such files can disrupt the operation of the system.

You cannot quarantine System Critical Objects (SCO). SCOs include files that are necessary for the operation of the operating system and the Kaspersky Embedded Systems Security application.

Quarantining a file is possible only if the integration of Kaspersky Embedded Systems Security with Kaspersky Managed Detection and Response is enabled and the MDR component is activated.

The directory for storing Quarantine objects must be writable.

Files are quarantined in one of the following ways:

- As a result of the execution of a task configured on the side of Kaspersky Managed Detection and Response.
- As a result of the execution of a command to manage the quarantine on the device.

For more information on working with Quarantine when integrated with Kaspersky Endpoint Detection and Response, see the Kaspersky Endpoint Detection and Response Help.

Managing quarantined files

You can manage quarantined files:

- In Kaspersky Security Center, in the common list of files quarantined by Kaspersky applications.

  To manage quarantined files in Kaspersky Security Center, you need to enable the transfer of data about quarantined files to the Administration Server.

- Locally on the device using the command line.

You can view information about quarantined files, and delete and restore files from quarantine.

Restoring or deleting a file from Quarantine is available regardless of whether integration with the Kaspersky Managed Detection and Response solution is enabled, and regardless of whether a policy is applied to the device.

The general list of files quarantined by Kaspersky applications on client devices is kept in Kaspersky Security Center and is available in the Administration Console (**Advanced** → **Repositories** → **Quarantine**) and the Web Console (**Operations** → **Repositories** → **Quarantine**). Kaspersky Security Center does not copy files from Quarantine storages to the Administration Server; all files are stored in Quarantine storages on client devices. For detailed information about managing quarantined files in Kaspersky Security Center, refer to the Kaspersky Security Center Help.

The quarantined file is restored to its original location according to the specified settings. Restoration preserves the MAC label (only on Astra Linux devices) and attributes of the original file (including the immutable and append-only attributes). A file with immutable and append-only attributes can only be restored to the location if the original file is not present, or to the original location, but with a different name. Root privileges are required to restore a file from Quarantine to a different directory or to the original location with a different name.

Once the restoration process is complete, the application deletes the quarantined copy of the restored file.

Restoring a file from quarantine fails in the following cases:

- The file to be restored was not found in the quarantine storage.
- The name of the file being restored is specified incorrectly or with the wrong case.
- The file ID was specified incorrectly.
- The destination directory has been deleted, moved, renamed, or you do not have access rights to it.

  In this case, the application moves the file to the /var/opt/kaspersky/kess/common/restored/ directory. You can manually move the file from this directory to the directory of your choice.
- A file with the same name already exists at the specified path.
- The device does not have enough space.
- Only on Astra Linux devices: the MAC label of the file is higher than the MAC label of the directory into which the file is being restored.
- Only on Astra Linux devices: the MAC label of the file is lower the MAC label of the directory into which the file is being restored, and the ccnr flag is not set for this directory.

Deleting a file from quarantine fails in the following cases:

- The file to be deleted was not found in the quarantine storage.
- The name of the file being deleted is specified incorrectly or with the wrong case.
- The file ID was specified incorrectly.

Editing Quarantine settings in the Web Console

In the Web Console, you can manage Quarantine settings in the policy properties (**Application settings** → **General Settings** → **Reports and Storage**) and in managed device properties (**Assets (Devices)** → **Managed Devices** → <device name> link → **Applications** tab → <Kaspersky Embedded Systems Security application name> link → **Application settings** tab → **Storage settings** section).

Quarantine settings

Setting	Description
Maximum size of Quarantine (MB)	If this check box is selected, the specified value sets the maximum quarantine size (in megabytes). If the check box is cleared, the size of quarantine is not limited. Possible values: 1–4000. Default value: 200.

Setting	Description
Notify when quarantine is (%) full:	If this check box is selected, an event is generated when Quarantine is filled to the specified percentage. If the check box is cleared, the event about Quarantine being full is not generated. Possible values: 1–100. Default value: 90.

Editing Quarantine settings in the Administration Console

In the Administration Console, you can manage Quarantine settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Storage settings**) and in managed device properties (**Managed devices** → **Devices** tab → device properties window → **Applications** section → application properties window → **Storage settings**).

Quarantine settings

Setting	Description
Limit the size of Quarantine to (MB):	This check box enables or disables limiting the maximum size of the Quarantine to the specified value (in megabytes). If the check box is cleared, the size of quarantine is not limited. Possible values: 1–4000. Default value: 200.
Notify when quarantine is (%) full:	This check box enables or disables the limit on the percentage of Quarantine that must be full to generate an event about Quarantine being full. If the check box is cleared, the event about Quarantine being full is not generated. Possible values: 1–100. Default value: 90.

Editing Quarantine settings on the command line

On the command line, you can configure Quarantine using the `QuarantineSizeLimit`, `QuarantineFillingNotification`, and `QuarantineBackupFolder` settings in the general application settings.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

Quarantine settings

Setting	Description	Value
`QuarantineFillingNotification`	The percentage of Quarantine that must be full to generate an event about Quarantine being full. To disable generating events about Quarantine being full, enter 0.	0 to 100 0 – generating events about Quarantine being full is disabled. Default value: 90.
`QuarantineSizeLimit`	Maximum size of the Quarantine (in megabytes). When the maximum Quarantine size is reached, the application deletes the oldest files. To make the size of Quarantine unlimited, enter 0.	0 to 4000 0 means the Quarantine size is not limited. Default value: 200.
`QuarantineBackupFolder`	Path to the directory in which the Backup and Quarantine objects are stored. You can specify a directory that is different from the default directory. You can use a directory on any device as the directory for storing Backup and Quarantine objects. It is not recommended to assign directories that are located on remote devices, such as those mounted via the Samba and NFS protocols. If the specified directory does not exist or is unavailable, the application uses the default directory.	Default value: /var/opt/kaspersky/kess/common/objects-backup/ Root privileges are required for access to the default directory.

Managing quarantined files on the command line

When integrated with Detection and Response solutions, on the command line, you can use Quarantine management commands to:

- Quarantine files.
- View information about quarantined files.
- Delete all files or specified files from Quarantine.
- Restore files from Quarantine.

Restoring infected files may lead to a device infection.

Quarantine file

To quarantine a file, run the following command:

```
kess-control [-Q] --put <file path> [--md5] [--sha256] [--save-original-file]
```

where:

- `<file path>` is the path to the file that you want to quarantine.
- `--md5` is the MD5 hash of the file that you want to quarantine.
- `--sha256` is the SHA256 hash of the file that you want to quarantine.
- `--save-original-file` keeps the original file. If you do not specify this option, the original file is deleted.

  The original file is deleted only if the user account has permission to delete objects. If the rights are insufficient, a file deletion error message is displayed.

Viewing information about quarantined files

To view information about quarantined files, run the following command:

```
kess-control -Q --query ["<filter conditions>"] [-n <number>] [--json]
```

where:

- `< filter conditions >` specifies one or more logical expressions in the `< field > < comparison operation > '< value >'` format combined using the `and` logical operator to restrict query results. If you do not specify any filter conditions, the application will display the details of all quarantined files.
- `< number >` is the count of the most recent quarantined files that you want to display. If you do not specify the `-n` option, the last 30 files are displayed. Specify 0 to show all files.
- `--json`: output data in JSON format.

The `ObjectId` line displays the numeric ID that the application assigned to the file when quarantining it. This ID is used to perform actions on the file, such as restoring the file or deleting it from Quarantine.

Restore files from Quarantine

To restore a file from Quarantine under its original name to its original location, execute the following command:

```
kess-control -Q --restore < object ID >
```

where `< object ID >` is a numeric ID that the application assigned to the file when it was quarantined.

To restore a file from Quarantine under a new name to a specified directory, execute the following command:

```
kess-control -Q --restore < object ID > --file < file path >
```

where `--file < file path >` is the new name of the file and the path to the directory where you want to save the file.

You can restore a file from Quarantine to a different directory only if your user has root rights. If your user's rights are insufficient, a file can only be restored to its original location.

On Astra Linux devices in the closed software environment mode, you can restore a file only under the same account under which it was placed in Backup.

Delete files from Quarantine

To delete selected files from Quarantine, run the following command:

```
kess-control -Q --mass-remove --query "< filter conditions >"
```

`where < filter conditions >` specifies one or more logical expressions in the `< field > < comparison operation > '< value >'` format combined using the `and` logical operator to restrict query results.

Examples:

To delete files that contain "test" in their names or paths:

```
kess-control -Q --mass-remove --query "FileName like '%test%'"
```

To delete all files from Quarantine, run the following command:

```
kess-control -Q --mass-remove
```

Viewing events and reports

While the application is running, various *events* occur. The events may be informational or may contain important data. For example, the application can use an event to inform about a successful application database update, or about an error of application components that needs to be fixed.

Kaspersky Embedded Systems Security saves information about application events to the following logs:

- The application event log. By default, the application saves information about events to the /var/opt/kaspersky/kess/private/storage/events.db database. You can configure the application event log in the command line.
- Operating system log (syslog). The operating system log is not used by default. You can enable saving events to this log.

Access to the application event log and operating system log requires root privileges.

If Kaspersky Embedded Systems Security is managed by Kaspersky Security Center, information about events may be transmitted to the Kaspersky Security Center Administration Server. Aggregation rules apply to certain events. If a large number of same-type events are created within a short period of time while the application is running, the application will switch to event aggregation mode and send to Kaspersky Security Center one aggregated event with a description of the events settings. Different aggregation rules may be used for different events. For more information about events, refer to the Kaspersky Security Center Help.

You can receive information about application events in the following ways:

- In the Administration Console and in the Web Console
- On the command line (Viewing events on the command line, Viewing events in Kaspersky Security Center)
- When using the application's graphical user interface – in pop-up notifications if displaying notifications is enabled in the policy properties or in the application's general settings

The application can generate various types of *reports* on the events that occur while the application is running. Reports contain information about the operation of each Kaspersky Embedded Systems Security component, the results of each task, and the overall operation of the application.

You can view reports in the following ways:

- Kaspersky Security Center reports are available in the Administration Console and in the Web Console. You can use these to get information about infected files or usage of keys and application databases, among other things. For detailed information on working with Kaspersky Security Center reports, please refer to the Kaspersky Security Center Help.
- The graphical user interface of Kaspersky Embedded Systems Security displays reports of the application, its components and tasks.

Contents of reports and events

Events and reports may contain the following personal data:

- User name and user ID of operating system users
- Paths to user files
- IP addresses of remote devices that are scanned by the Anti-Cryptor component
- IP addresses of senders and receivers of network packets scanned by the Firewall Management component
- Web addresses of the update sources
- General application settings values
- Names and settings of command line tasks
- Detected malicious, phishing, adware web addresses, and web addresses containing legitimate applications that intruders can use to compromise devices or data
- Names and IDs of the devices
- Web addresses of the repositories
- File names, paths to files, and hash-sums of executable application files
- Names of application categories
- Web addresses of the repositories
- File names, paths, and hashes of executable files of applications

Rules of escaping characters in file paths

Some events may contain file paths. For output, the file path is treated as a UTF-8 string. If any of the bytes in the path does not comply with the UTF-8 encoding rules, is it replaced with the `?` character. Any four-byte sequence that encodes a character code outside the Unicode range (greater than 0x10FFFF) is also replaced with the `?` character. Special characters are escaped (replaced) in a certain way.

The following rules apply for escaping characters in file paths inside events in the output of `kess-control -E --query`:

- '\a', '\b', '\t', '\n', '\v', '\f', '\r' characters are replaced by two characters as follows:
  '\a' -> "\\a"
  '\b' -> "\\b"
  '\t' -> "\\t"
  '\n' -> "\\n"

'\v' -> "\\v"

'\f' -> "\\f"

'\r' -> "\\r"

- All other special characters are output without modification.

The following rules apply for escaping characters in file paths inside events in the output of `kess-control -E --query --json`:

- In accordance with the JSON format, the '\b', '\f', '\n', '\r', '\t', '"', '\\' characters are escaped as follows:

  '\b' -> "\\b"

  '\f' -> "\\f"

  '\n' -> "\\n"

  '\r' -> "\\r"

  '\t' -> "\\t"

  '"' -> "\\\""

  '\\' -> "\\\\"

- All other special characters are escaped in accordance with the general JSON rules for escaping special characters ('\a' -> '\u0007').

Rules for escaping characters in file paths in events when sending to syslog:

- In accordance with the JSON format, the '\b', '\f', '\n', '\r', '\t', '"', '\\' characters are escaped as follows:

  '\b' -> "\\b"

  '\f' -> "\\f"

  '\n' -> "\\n"

  '\r' -> "\\r"

  '\t' -> "\\t"

  '"' -> "\\\""

  '\\' -> "\\\\"

- All other special characters are escaped in accordance with the general JSON rules for escaping special characters ('\a' -> '\u0007').

The first backslash in the sequence when describing rules is the escape character.

Examples:
`'\a'` is one character (a control character).

`'\\a'` is two characters (backslash + the `a` character).

`'\\'` is one character (backslash), `'\\\\'` is two characters (backslash + backslash).

Configuring event logging to the operating system log

Events that occur during the operation of Kaspersky Embedded Systems Security can be recorded in the operating system log. You can enable or disable the recording of events in this log using the Web Console, Administration Console, or the command line.

If you use the command line to manage the application, the recording of events in the operating system log is disabled by default.

If you use Kaspersky Security Center to manage the application, by default, information about adding and removing license keys and about license term expiration is recorded in the operating system log. You can select the events that you want to be recorded in the operating system log in the properties of the policy that is applied to the application.

Configuring in the Web Console

In the Web Console, you can enable or disable event logging to the operating system log in the policy properties (**Application settings** → **General Settings** → **Event settings**).

Clicking the **Settings** button under the event severity level (critical events, functional failures, warnings, informational messages) opens a window containing a list of events of the corresponding severity level. For each event, you can configure logging in the operating system log.

Configuring in the Administration Console

In the Administration Console, you can configure the logging of events to the operating system log in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**).

Clicking **Configure** under **Notifications** opens the **Notification settings** window. In this window, you can use the check boxes to select the events that the application records in the operating system log.

You can select individual event types or all event types of a certain severity level (critical events, functional failures, warnings, informational messages).

Configuring on the command line

You can enable or disable saving events to the operating system log in the command line via the `UseSyslog` option in the general application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `UseSyslog` parameter can take the following values:

- `Yes`: enable saving events to syslog.
- `No` (default): disable saving events to syslog.

Configuring application event log settings

By default, information about events is saved to the application event log located on the device.

The event log database contains the following files:

- Several files with events. The number of files depends on the current number of events in the event log and on the maximum number of events that the application is configured to store (the `MaxEventsNumber` setting).

  The name of the file with events includes its serial number (for example, events.db.0001). When a file fills up, a new file is created. If the maximum number of events in all event files not counting the earliest file reaches the `MaxEventsNumber` value, the earliest file is deleted. This means that the application may store more events than indicated in the setting.

- The main file of the database for managing event files that stores information about the event log files. This file is not deleted.

You can manage the following application event log options on the command line via the general application settings:

- Change the path to the application event log database using the `EventsStoragePath` setting. Default value: /var/opt/kaspersky/kess/private/storage/events.db.

- Specify the maximum number of events to be stored by the application via the `MaxEventsNumber` option. Default value: 500000.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

Viewing events in Kaspersky Security Center

A list of all Kaspersky Embedded Systems Security events is displayed in the Web Console and in the Administration Console.

Events of the Kaspersky Embedded Systems Security application fall into the following types by severity:

- *Critical events* pose a serious threat to the device's security (detection of malicious objects, vulnerabilities, problems with the operation of the application). Critical events require your immediate attention. We recommend not to disable critical event notifications.

- *Functional failures* occur when the application fails (processing error, task misconfiguration). Such events do not pose a direct threat to the security of the device, but they require your actions for the application to work correctly.

- *Warnings* do not require your immediate action, but may indicate a potential future threat to the security of the device.

- *Informational messages* correspond to events of an informational nature.

You can configure event notifications. A *notification* is a message containing information about an event that occurred on a protected device. Notifications give you timely information about application events. You can configure the execution of a script upon receiving events from the application or upon receiving notifications about events by e-mail.

For detailed information about using Kaspersky Security Center notifications, refer to the Kaspersky Security Center Help.

Viewing events on the command line

On the command line, you can view:

- Current application events
- Events in the application event log

Displaying current events

You can output information about all current application events or about current events associated with starting or stopping a specified task. You can use the filter to output certain current events, for example, events of a specified type.

To output information about all current application events, run:

```
kess-control -W
```

The command returns the name of the event and additional information about the event.

To output only information about current events associated with a running task, run:

```
kess-control --start-task <task ID/name> -W
```

Example:
Enable display of the current events of the running task with ID=1:

```
kess-control --start-task 1 -W
```

To output information about current events that match the filter conditions, run:

```
kess-control -W --query "<filter conditions>"
```

Filter conditions are set with one or more logical expressions in the format `<field> <comparison operator> '<value>'`, combined with the logical operator `and`.

Example:
Display TaskStateChanged events:

```
kess-control -W --query "EventType == 'TaskStateChanged'"
```

Display TaskSettingsChanged events initiated by the 'User' user:

```
kess-control -W --query "EventType == 'TaskSettingsChanged' and Initiator == 'User'"
```

Displaying events from the event log

You can output information about events from the application event log to the console or a file. You can use a filter to display only certain events.

To output information about all events in the application event log, run:

```
kess-control -E --query [--db <database file>]
```

where <database file> is the full path to the event log database file to output events from. By default, the application saves information about events to the /var/opt/kaspersky/kess/private/storage/events.db database. The location of the database is determined by the EventsStoragePath general application setting.

If the event log is located in the default database, you can output information about all events using the kess-control -E command.

You can use less to navigate the list of displayed events. By default, the application stores up to 500,000 events. The maximum number of events that the application stores is determined by the MaxEventsNumber general application setting.

To output information about events in the application event log that meet certain criteria, run:

```
kess-control -E --query "<filter conditions>" [--db <database file>] [-n <number>] [--json] [--reverse]
```

where:

- <filter conditions> specifies one or more logical expressions in the <field> <comparison operation> '<value>' format combined using the and logical operator to restrict query results.
- <number> is the number of the latest events of the selection (records counting from the end of the selection) to be displayed.
- --json: output events in JSON format.
- --reverse: display events in reverse order (from the newest event at the top to the oldest at the bottom).

To output information about events in the application event log that meet certain criteria to a file, run:

```
kess-control -E --query "<filter conditions>" [--db <database file>] [-n <number>] --file <file path> [--json]
```

where `--file <file path>` is the full path to the file to output events to.

Configuring pop-up notifications in the graphical interface

By default, in the graphical interface of the application, information about events is displayed in pop-up notifications. You can enable or disable pop-up notifications using the Administration Console or the command line.

Configuring in the Administration Console

In the Administration Console, you can enable or disable the display of pop-up notifications in the graphical interface in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**General settings** → **Application settings**).

The **Show pop-up notifications in the graphical user interface** check box enables pop-up notifications in the graphical user interface.

This check box is selected by default.

Configuring on the command line

On the command line, you can enable or disable pop-up notifications in the graphical interface using the `ShowPopUpNotifications` setting in the General application settings.

You can edit the option via command line switches or a configuration file that contains all general application settings.

The `ShowPopUpNotifications` parameter can take the following values:

- `Yes` (default value) – show pop-up notifications in the graphical user interface.
- `No` – do not show pop-up notifications in the graphical user interface.

Application management via the graphical interface

You can do the following in the Kaspersky Embedded Systems Security graphical user interface:

- View information about device protection status.
- Enable and disable application components:
  - File Threat Protection.
  - Firewall Management.
  - Anti-Cryptor.
  - Removable Drives Scan.
  - Web Threat Protection.
  - Network Threat Protection.
  - Device Control.
  - BadUSB Attack Prevention.
  - Behavior Detection.
  - Application Control.
  - System Integrity Monitoring.
- Start and stop scan tasks:
  - Malware Scan.
  - Critical Areas Scan.
- Run a custom scan of files and directories.
- Start and stop database update and rollback tasks.
- Enable and disable Kaspersky Security Network.
- Manage application license keys and view information about the license under which the application is being used as well as the key associated with the license.
- Request and obtain temporary access to a blocked device from the administrator of the organization's local network.
- View application statistics and reports.

- View information about objects placed in Backup.
- Create application trace files.

If an application component and/or task is running in "Notify only" mode, the GUI of the component or task displays the warning *"Notify only" mode selected*.

Graphical user interface

Application icon in the notification area

The Kaspersky Embedded Systems Security icon in the tray provides access to the context menu and the main window of the graphical interface of the application.

> In some operating systems, the default application icon may be missing after installing the application graphical interface package.

The context menu of the application icon contains the following items:

- **Kaspersky Embedded Systems Security 4.0 for Linux**. Opens the main window of the graphical interface of the application, which displays the protection status of a device and contains interface elements that provide access to the application functions.
- **Exit**. Exits the graphical interface.

The main window of the application graphical interface

To open the main application window, do one of the following:

- Select the application name in the application menu of the operating system window manager.
- Right-click or double-click the application icon in the notification area of the taskbar.

The main graphical interface window of the application is divided into several parts:

- The upper part of the main window displays the protection status of the device. Clicking the **View** button opens the **Protection Center** window. This window displays information about the protection status of a device and recommendations on the actions to be performed to fix protection problems (if any).
- Under **Scan**, the Malware Scan task status and the number of detected threats are displayed. Clicking here opens the **Scan** window. In this window, you can start and stop the *Malware Scan* and *Critical Areas Scan* tasks. Also, you can view reports for these tasks.
- Under **Update**, the status of the *Update* task is displayed. Clicking here opens the **Update** window. In this window, you can start the *Update* and *Rollback* tasks. Also, you can view reports for these tasks.

- The lower part of the main window of the application contains the following elements:
  - **Reports** button. Clicking this button opens the **Reports** window, where you can view component and task statistics and various reports.
  - **Backup** button. Clicking this button opens the **Backup** window, which contains information about objects in Backup.
  - **Settings** button. Clicking this button opens the **Settings** window, where you can enable or disable application components and configure the use of the Kaspersky Security Network.
  - **Temporary access** button. Clicking this button opens the **Temporary access** window, in which you can request and obtain temporary access to the blocked device.
  - **Support** button. Clicking this button opens the **Support** window where you can view information about the application and create an application trace file.
- The lowest part of the main application window displays information about the license and the key, as well as about licensing problems (if any). Clicking on this area of the window opens the **Licenses** window, which displays detailed license information. Clicking the **Purchase a license** button in this window opens the Kaspersky online store, where you can purchase a license. After purchasing a license, you will receive an activation code or a key file, which you must use to activate the application.

Managing the graphical interface requires the rights of the Administrator (admin) role. Parts of the graphical interface may not be available to users with other roles.

Enabling and disabling application components

You can enable or disable application components using a toggle switch in the graphical interface.

By default, the following components are enabled: File Threat Protection, Device Control, and Behavior Detection. The Web Threat Protection component can be enabled automatically if local management of Web Threat Protection settings is allowed on the device (a policy is not applied or the "lock" is not set in the policy properties) and one of the supported browsers is detected in the system.

To enable or disable an application component:

1. Open the main window of the application graphical interface.
2. In the lower part of the main window, click the **Settings** button.

   The **Settings** window opens.
3. Enable or disable the toggle switch for the relevant component.

Starting and stopping scan tasks

To start or stop a scan task:

1. Open the main window of the application graphical interface.
2. In the main window, click **Scan**.

   The **Scan** window will open.
3. Do one of the following:

   - To start a scan task, click the **Start** button under the scan task that you want to start.

     The progress of the running scan task is displayed.
   - To stop a scan task, click the **Stop** button under the scan task that you want to stop.

     The scan task stops, and information about the scanned objects and detected threats is displayed.
4. To view the report on the scan task, click the **Show report** button.

When an infected object is detected or the scan task is completed, a pop-up window appears in the notification area near the application icon on the right side of the taskbar.

The **Scan** window also displays the progress and results of temporary boot sector scan tasks (*Scan_Boot_Sectors_{ID}*) and temporary custom file scan tasks (*Scan_File_{ID}*). You can hide information about temporary tasks that are already completed by clicking the cross or by closing the **Scan** window (when switching to the main window or when exiting the application).

Starting a custom scan

You can use the graphical user interface to run a custom scan of files and directories.

To run a custom scan of a file or directory:

1. Right-click the file or directory that you want to scan to open the context menu.
2. Select **Kaspersky Embedded Systems Security 4.0 for Linux** as the application for scanning the file or directory.

   Depending on the desktop environment, the names of the context menu commands may differ.

The custom scan task and its progress are displayed in the graphical user interface.

Starting and stopping the Update task

To start or stop an update task:

1. Open the main window of the application graphical interface.
2. In the main application window, click **Update**.

   The **Update** window opens.
3. Do one of the following:

   - To start a task, click the **Start** button under the task that you want to start.

     The progress of the running update task is displayed.

     If the Update task finishes successfully, the **Roll back update** link becomes available, and you can roll back the last successful database update.
   - To stop a task, click the **Stop** button under the scan task that you want to stop.

     The Update task stops.
4. To view the report on the task, click the **Show report** button.

To start a rollback task:

1. Open the main window of the application graphical interface.
2. In the main application window, select the **Update** section.

   The **Update** window opens.
3. Run the Rollback task by clicking the **Roll back update** link.

Configuring Kaspersky Security Network

You can enable or disable Kaspersky Security Network using the graphical interface.

To enable use of Kaspersky Security Network:

1. Open the main window of the application graphical interface.
2. In the lower part of the main window, click the **Settings** button.

   The **Settings** window opens.
3. In the **Settings** window, select one of the following options:

   - **Extended KSN mode**, if you want to use Kaspersky Security Network, obtain information from the knowledge base, and send anonymous statistics and information about the types and sources of threats.
   - **Basic KSN mode**, if you want to use Kaspersky Security Network, obtain information from the knowledge base, but not to send anonymous statistics and information about the types and sources of threats.

4. Click the **Enable** button.

   The **Using Kaspersky Security Network** window opens.

5. In the **Using Kaspersky Security Network** window, carefully read the Kaspersky Security Network Statement and select the **I confirm that I have fully read, understand, and accept the terms and conditions of the Kaspersky Security Network Statement** option.

6. Click **OK**.

   The **OK** button is not available if none of the options are selected in the **Kaspersky Security Network usage** window.

To disable use of Kaspersky Security Network:

1. Open the main window of the application graphical interface.

2. In the lower part of the main application window, click the **Settings** button.

   The **Settings** window opens.

3. Click **Enable**.

4. In the window that opens, click the **Yes** button to decline use of Kaspersky Security Network.

Viewing reports

You can use the graphical user interface to view application reports. Reports contain information about the operation of application components and tasks. Reports are displayed in the window that opens when clicking **Reports** in the lower part of the main window of the application graphical interface.

Report data is presented as a table that contains a list of events. Each line in the table contains information about a separate event. Event attributes are displayed in the table columns. Events logged while various components and tasks are running have different sets of attributes.

The following event importance levels are used in reports:

- *Critical* – events with the critical importance level that need your attention as they indicate problems in the application operation or vulnerabilities in the device protection
- *High*
- *Medium*
- *Low*
- *Information*
- *Error*

The following reports are available in the application:

- **Statistics**. This report contains File Threat Protection and scan task statistics. You can update the displayed report by clicking the **Reload** button.
- **System audit**. This report contains information about events that occur during the application operation and during the user interaction with the application.
- **Threat protection**. This report contains information about the events that are logged while the following application components were running:
  - File Threat Protection
  - Removable Drives Scan
  - Anti-Cryptor
  - Web Threat Protection
  - Network Threat Protection
  - BadUSB Attack Prevention
  - Firewall Management
  - Behavior Detection
  - Application Control
  - Device Control
  - System Integrity Monitoring
- **On-demand tasks**. This report contains information about events logged by scan tasks, update tasks, and system integrity check tasks.

To view the report:

1. Open the main application window.
2. In the lower part of the main application window, click the **Reports** button.

   The **Reports** window will open.
3. In the left part of the **Reports** window, select the required report type.

   A report containing a list of events is displayed in the right part of the window.

   By default, events are sorted in ascending order by the values in the **Date** column.
4. To view detailed information about an event, select the event in the report.

   The section that contains the attributes of this event is displayed at the bottom of the window.

For convenient processing of reports, you can modify the presentation of data on the screen in the following ways:

- Filter the list of events by occurrence time.
- Use the search function to find a specific event.
- View the selected event in a separate section.

Viewing Backup objects

You can do the following with objects in Backup using the graphical interface:

- View information about objects placed in Backup on the device.
- Restore objects from Backup to their original directories.
- Remove objects from Backup. Deleted objects cannot be restored in the future.

Information about restoring and deleting objects is saved to the event log.

To view objects in Backup:

1. Open the main window of the application graphical interface.
2. In the lower part of the main application window, click the **Backup** button.
   This opens the **Backup** window.

In this window the following information about the objects in Backup storage is displayed:

- Object name.
- Full path to an object.
- The date when the object was added to Backup.
- The date when the object was deleted from Backup (this field is displayed if Backup retention period is set).
- Object size.

Managing license keys

Using the graphical user interface, you can add and remove application license keys, and view information about the license under which the application is being used as well as the associated license key.

You can activate the application by adding an active license key.

Activation is the process of activating a license that allows you to use a fully functional version of the application until the license expires.

To activate the Kaspersky Embedded Systems Security application on a protected device, you need to add a license key to the application. Also, a separate license key is provided for the Kaspersky Embedded Systems Security application that can be used to activate the integration with Kaspersky Managed Detection and Response; this is called the MDR key.

You can also add a reserve key for the basic functionality and a reserve key for the additional functionality. A reserve key becomes active when the license associated with the active key expires or when the active key is deleted. Availability of a reserve key allows you to avoid application functionality limitation when your license expires.

A reserve license key can be added only after adding an active license key.

Adding a license key

If the application is not yet activated (an active main license key has not been added), you can activate the application under a commercial license or under a trial license.

To activate the application:

1. Open the main window of the application graphical interface.
2. Do one of the following:
   - In the lower part of the main window, click the area that displays information about the license and the key.
   - In the lower part of the main window, click the **Support** button to open the **Support** window, and in that window, click either the **License key** or the **Licenses** link in the lower part of the window.

   This opens the **Licenses** window. Clicking the **Purchase a license** button in this window opens the Kaspersky online store, where you can purchase a license.
3. If you want to activate the application under a commercial license:
   a. Click the **Add** button in the **Commercial key** section and perform the following actions, depending on the method you are using to add the key:
      - If you want to add a key using an activation code, enter the activation code and click the **Next** button.
      - If you want to add a key using a key file, click the **Select a key file** link to open the file selection window that window, select the file with the .key extension.

      The window displays information about the key and the license associated with it.
   b. Click the **Activate** button.
4. If you want to activate the application under a trial license, click the **Activate** button in the **Trial key** section. The window displays information about the trial license and the associated key.

   You can use the application under a trial license for only one trial period.

After activating the application, you can add a reserve main license key and, if necessary, an active MDR key and a reserve MDR key. The procedures for adding a reserve main key and a MDR key are the same.

To add a reserve main key or MDR key to the application:

1. Open the main window of the application graphical interface.
2. Do one of the following:
   - In the lower part of the main window, click the area that displays information about the license and the key.
   - In the lower part of the main window, click the **Support** button to open the **Support** window, and in that window, click either the **License key** or the **Licenses** link in the lower part of the window.

   This opens the **Licenses** window. The window displays information about previously added license keys.
3. Click the **Add** button and perform the following actions, depending on the method you are using to add the key:
   - If you want to add a key using an activation code, enter the activation code and click the **Next** button.
   - If you want to add a key using a key file, click the **Select a key file** link to open the file selection window that window, select the file with the .key extension.

   The window displays information about the key and the license associated with it.
4. Click the **Add** button.

If you add a main license key and the application is already activated, the new key is added as a reserve main key.

If you add an MDR key, it is added as an active key if an MDR key has not been added to the application previously. The new MDR key is added as a reserve key if an MDR key already has been added to the application.

> If both an active key and a reserve key were previously added to the application, the new key replaces the previously added reserve key.

A trial license key and a subscription key cannot be added as a reserve key.

Removing a license key

To remove a license key that has been added to the application:

1. Open the main window of the application graphical interface.
2. Do one of the following:
   - In the lower part of the main window, click the area that displays information about the license and the key.
   - In the lower part of the main window, click the **Support** button to open the **Support** window, and in that window, click either the **License key** or the **Licenses** link in the lower part of the window.

   This opens the **Licenses** window.

3. Click the **Remove** button to the right of the information about the key that you want to remove.

4. Confirm the removal in the window that opens.

Viewing license information

To view license information:

1. Open the main window of the application graphical interface.

2. Do one of the following:

   - In the lower part of the main window, click the area that displays information about the license and the key.

   - In the lower part of the main window, click the **Support** button to open the **Support** window, and in that window, click either the **License key** or the **Licenses** link in the lower part of the window.

   This opens the **Licenses** window.

   The window displays information about the active and reserve keys added to the application. Click the **More info** link to view full information about the licenses and keys.

The **Current licenses** section displays information about the added active keys (active main key and active additional functionality key) and the licenses associated with these keys:

- License type, license limitation, and license term.
- **License key** – a unique alphanumeric sequence.
- **Key status** – the status of the key or a message about the problems associated with the key (if any).
- **Valid from** – date when the application was activated by adding this key.
- **Expires** – the number of days before the license expires and the license expiration date (UTC).
- **Name** – the name of the application for which the activation key was added.
- **Protection** – information about restrictions on protection functions and the ability to update application databases.

The **Reserve keys** section displays information about the added reserve keys (reserve main key and reserve additional functionality key) and the licenses associated with these keys:

- Type of reserve key, license limit, and license term associated with the key.
- **License key** – a unique alphanumeric sequence.
- **License type** – the type of license associated with the reserve key.
- **Name** – the name of the application for which the activation key was added.
- **Protection** – information about restrictions on protection functions and the ability to update application databases.

Gaining temporary access to device

When a device is connected, and access to that device is denied by Device Control, a window with a message about blocked access to the device is displayed.

You can request temporary access to a blocked device from your organization's local network administrator using the graphical interface:

- From the window with the message about blocked access to the device, by clicking **OK**
- From the main window of the application graphical interface

To request temporary access to a blocked device:

1. Open the main window of the application graphical interface.
2. In the lower part of the main window, click the **Temporary access** button.

   This opens the **Temporary access** window.

   You can also open the **Temporary access** window from the window with the message about blocked access to the device by clicking **OK**.
3. In the **Request access to the blocked device** section, in the list of connected devices, select the device to which you want to gain access.
4. In the **Device access duration (hours)** field, specify the time period for which you want to gain access to the device (in hours).
5. Click the **Create request access file** button.
6. In the opened window, specify the path to the directory in which you want to save the access request file with the .akey extension.
7. Send the access request file to the administrator, for example, by email or in any available way (the administrator can configure this in the block message template).

After getting the access key from the administrator, you can upload the access key to gain temporary access to the blocked device.

To get temporary access to a blocked device:

1. Open the main window of the application graphical interface.
2. In the lower part of the main window, click the **Temporary access** button.

   This opens the **Temporary access** window.
3. In the **Get access to the device** section, click the **Upload access key** button.
4. In the opened window, select the file with the device access key (file with the .acode extension) received from the administrator of the organization's local network.

5. A window is displayed with information about the device to which access has been granted and the access duration.

6. Click **OK**.

Viewing information about the application

You can view information about the installed application and its device in the **Support** window, which you can open by clicking the **Support** button in the lower part of the main window of the application graphical interface.

The **Support** window displays the following information:

- Application version.
- **License key** displays the active main license key added to the application, or a message saying that no key has been added. The link in this field opens the **Licenses** window, which displays detailed license information.
- **Key status** displays information about the status of the active main license key, or a message saying that no key has been added.
- **Database release date** – status and release date of the application databases.
- **Operating system** – information about the operating system of the device.
- **Application management** – Information about whether the device is managed by a policy.

If the **Application is allowed to retrieve policy's extended data on device** option is enabled in the Network Agent policy, the window also displays the following information about the policy that is applied on the device with the application installed.

- **Policy name** is the name of the policy that is applied to the device on which application is installed.
- **Date applied** – the date and time when the policy was last applied or updated on the device with the installed application.
- **Administration group** is the name of the administration group that includes the device on which the application is installed, that is managed by a policy.
- **Active policy profiles** is a list of active policy profiles applied on the device (if any).

The following links are displayed at the bottom of the **Support** window:

- **Licenses**, which opens the **Licenses** window, which displays detailed license information.
- **Knowledge Base**, which opens the Knowledge Base page.
- **Support**, which opens the Kaspersky CompanyAccount page.
- **Tracing**, which opens the **Tracing** window. In this window, you can create application trace files and configure the level of detail of the trace files.

  To follow the **Tracing** link, you need the rights of the admin role.

Creating a trace file

You can create application trace files and configure the level of detail of the trace files using the graphical interface.

To create a trace file and configure its level of detail, you need a user with the Administrator role (admin).

To create a trace file:

1. Open the main window of the application graphical interface.
2. In the lower part of the main application window, click the **Support** button.

   The **Support** window opens.
3. Click the **Tracing** link to open the **Tracing** window
4. In the **Level** drop-down list, select the level of detail for the trace file.

   You are advised to clarify the required level of detail with a Kaspersky Technical Support specialists. The default value is **Diagnostic (300)**.
5. Click the **Enable** button to start tracing.
6. Reproduce the situation which caused the problem.
7. Click the **Disable** button to stop tracing.

Created trace files are stored in the /var/log/kaspersky/kess/ directory.

Application components integrity check

Kaspersky Embedded Systems Security contains many various binary modules in the form of dynamic linked libraries, executable files, configuration files, and interface files. A hacker can replace one or more executable modules or files of the application with other files containing malicious code. To prevent such replacement of modules and files, Kaspersky Embedded Systems Security can check the integrity of application components. The application checks modules and files for unauthorized changes or corruption. If an application module or file has an incorrect checksum, it is considered to be corrupted.

An integrity check is run for the following application components if installed on the device:

- application package
- Graphical user interface package
- Kaspersky Security Center Network Agent package
- Kaspersky Embedded Systems Security administration web plug-in
- Kaspersky Embedded Systems Security administration MMC plug-in

The integrity of the application components is checked using an integrity check utility. The utility checks integrity of the files in the special lists called *manifest files*. Each application component has its own manifest file that contains a list of application files whose integrity is important for correct operation of this application component. The name of the manifest file is the same for each component, but the content of the manifest files differs. The manifest files are digitally signed and their integrity is checked as well.

To run the integrity check utility on Linux devices, an account with root privileges is required. An administrator account is required to run the integrity check utility on Windows devices.

The integrity check utility is installed with the application and located at the following paths:

- To check the application package, graphical user interface package, and Network Agent: /opt/kaspersky/kess/bin/integrity_checker.
- To check the Kaspersky Embedded Systems Security administration MMC plug-in
  - %ProgramFiles%\Kaspersky Lab\Kaspersky Security Center\Plugins\kess_<version number>.plg\integrity_checker.exe – for 32-bit operating systems
  - %ProgramFiles(x86)%\Kaspersky Lab\Kaspersky Security Center\Plugins\kess_<version number>.plg\integrity_checker.exe – for 64-bit operating systems
- To check the Kaspersky Embedded Systems Security administration web plug-in:
  - %ProgramFiles%\Kaspersky Lab\Kaspersky Security Center Web Console\integrity_checker.exe – on devices with Windows operating systems
  - /var/opt/kaspersky/ksc-web-console/integrity_checker – on devices with Linux operating systems

The manifest files are located at the following paths:

- To check the integrity of the application package: /opt/kaspersky/kess/bin/integrity_check.xml.
- To check the integrity of the graphical user interface package: /opt/kaspersky/kess/bin/gui_integrity_check.xml.
- To check Network Agent:
  - /opt/kaspersky/klnagent/bin/kl_file_integrity_manifest.xml – for 32-bit operating systems
  - /opt/kaspersky/klnagent64/bin/kl_file_integrity_manifest.xml – for 64-bit operating systems
- To check the Kaspersky Embedded Systems Security administration MMC plug-in
  - %ProgramFiles%\Kaspersky Lab\Kaspersky Security Center\Plugins\kess_<plugin version>.plg\integrity_check.xml – for 32-bit operating systems
  - %ProgramFiles(x86)%\Kaspersky Lab\Kaspersky Security Center\Plugins\kess_<plugin version>.plg\integrity_check.xml – for 64-bit operating systems
- To check the Kaspersky Embedded Systems Security administration web plug-in:
  - %ProgramFiles%\Kaspersky Lab\Kaspersky Security Center Web Console\server\plugins\kess_<version number>\integrity_check.xml – on devices with Windows operating systems
  - /var/opt/kaspersky/ksc-web-console/server/plugins/kess_<version number>\integrity_check.xml – on devices with Linux operating systems

To check the integrity of a solution component, you need to run the tool from the folder of that component's tool.

To run the integrity check utility, run one of the following commands:

- To check the integrity of the application package and graphical user interface package:

  ```
  integrity_checker [<path to manifest file>] --signature-type kds-with-filename
  ```

- To check the integrity of the Kaspersky Embedded Systems Security administration MMC plug-in:

  ```
  integrity_checker.exe [<path to manifest file>]
  ```

- To check the integrity of the Kaspersky Embedded Systems Security administration web plug-in and Network Agent on Linux devices:

  ```
  integrity_checker [<path to manifest file>]
  ```

- To check the integrity of Kaspersky Embedded Systems Security administration web plug-ins on Windows devices:

  ```
  integrity_checker.exe [<path to manifest file>]
  ```

  The default path is for a manifest file located in the same directory as the integrity checker utility.

You can run the utility with the following optional settings:

- `--crl <directory>` is the path to the directory containing the Certificate Revocation List.
- `--version` – display the version of the utility.

- `--verbose` – display detailed information about performed actions and their results. If you do not specify this setting, only errors, objects that did not pass the check, and scan statistics summary will be displayed.

- `--trace < file name >`, where `< file name >` is the name of the file where events that happen during scans will be logged at the DEBUG level of detail.

- `--signature-type kds-with-filename` – the type of the signature to be checked (this setting is required for checking the application package, graphical user interface package, and Network Agent).

- `--single-file < file >` checks only the specified file in the manifest and ignores all the other objects of the manifest.

You can view description of all available integrity check utility settings in the help on the utility options by running the `integrity_checker --help` command.

The result of checking the manifest files is displayed as follows:

- `SUCCEEDED` — integrity of the files has been confirmed (return code 0).

- `FAILED` – integrity of the files has not been confirmed (return code is not 0).

If a violation of the integrity of the application or Network Agent is detected when the application starts, Kaspersky Embedded Systems Security generates the corresponding event in the event log and in Kaspersky Security Center.

Contact Technical Support

If you do not find a solution to your problem in the application documentation or other sources of information about the application, it is recommended to contact Technical Support. Technical Support specialists will answer any of your questions about installing and using Kaspersky Embedded Systems Security.

Kaspersky provides support for Kaspersky Embedded Systems Security during its life cycle (see the Application life cycle page). Before contacting Technical Support, please read the support rules.

You can contact Technical Support in one of the following ways:

- Visit Technical Support website.
- Submit a request to Kaspersky Technical Support from the Kaspersky CompanyAccount portal.

Technical Support via Kaspersky CompanyAccount

Kaspersky CompanyAccount is a portal for companies that use Kaspersky applications. The Kaspersky CompanyAccount portal is designed to facilitate interaction between users and Kaspersky specialists through online requests. The Kaspersky CompanyAccount portal lets you monitor the progress of electronic request processing by Kaspersky specialists and store a history of electronic requests.

You can register all of your organization's employees under a single account on Kaspersky CompanyAccount. A single account lets you centrally manage electronic requests from registered employees to Kaspersky and also manage the privileges of these employees via Kaspersky CompanyAccount.

The Kaspersky CompanyAccount portal is available in the following languages:

- English
- Spanish
- Italian
- German
- Polish
- Portuguese
- Russian
- French
- Japanese

To learn more about Kaspersky CompanyAccount, visit the Technical Support website.

Obtaining information for Technical Support

After you inform Kaspersky Technical Support specialists about the problem, they may ask you to send a *trace file* or *dump file*.

Technical Support specialists may also need additionally information about the operating system and running processes on the device, as well as detailed reports on the operation of application components.

While diagnosing the problem, Technical Support specialists may ask you to change the application settings to:

- activate functionality to receive advanced diagnostic information;
- perform more detailed configuration of individual application components that cannot be performed through the standard user interface;
- change settings for storing received diagnostic information;
- to configure the capture and storage of network traffic in a file.

Technical Support specialists will tell you all the information required to perform these actions (the sequence of steps, the settings to change, configuration files, scripts, advanced command line capabilities, debugging modules, special utilities, etc.), as well as the body of information received for diagnostic purposes. The received advanced diagnostic information is stored on the user device. This information is not automatically sent to Kaspersky.

The steps listed above should be performed only with the guidance of Technical Support specialists based on instructions they provide. Independently changing application files using means not described in the application documentation or not recommended by Technical Support specialists may lead to poor performance and failures in the application and operating system, reduced protection, as well as inaccessible and corrupted data.

About application trace files

A Kaspersky Embedded Systems Security *trace file* tracks the step-by-step execution of application commands and lets you find out at which stage the error occurs.

Application trace files are not generated by default. You can enable or disable generation of application trace files and define the level of detail in trace files in the command line via the general application settings and the graphical user interface.

If you have enabled the generation of application trace files, these files are saved in /var/log/kaspersky/kess/. Access to this directory requires root privileges.

To store trace files, the application needs disk space in addition to the disk space listed in minimum hardware requirements.

Trace files are stored on the device as long as the application is in use, and are deleted permanently when the application is removed. Trace files are not sent to Kaspersky automatically.

Trace files are saved in a human-readable format. It is recommended to protect information from unauthorized access before sending it to Kaspersky.

Application trace file contents

Trace files contain the following general data:

- Event time.
- Number of the thread of execution.
- Application component that caused the event.
- Degree of event severity (informational event, warning, critical event, error).
- A description of the event involving command execution by a component of the application and the result of execution of this command.

Trace files may store the following information in addition to general data:

- The statuses of the application components and their operational data.
- Data on user activity in the application.
- Data on the hardware installed on the device.
- Data about all operating system objects and events, including information about user activity.
- Data contained in the objects of the operating system (for example, the contents of files that may contain any user personal data).
- Network traffic data (for example, the contents of the entry fields on a website, which may include bank card information or any other sensitive data).
- Data received from Kaspersky servers (such as the version of the application databases).
- Data on consumed CPU resources.
- Data on consumed RAM resources.
- Data about file read and write operations by applications.
- Data on the amount of cached information required for the application to work.

Configuring application trace settings

If you are managing Kaspersky Embedded Systems Security using Kaspersky Security Center, you can configure trace settings of the application in the Kaspersky Embedded Systems Security policy settings using the Web Console or the Administration Console.

If you are managing the application on the command line, you can configure trace settings of the application in the general application settings.

Editing trace settings in the Web Console

In the Web Console, you can manage application trace settings in the policy properties (**Application settings** → General Settings → Technical Support, **Trace file settings** section).

Trace settings

Setting	Description
Trace folder path	Input field for the path to the directory where the trace files are stored. Default value: /var/log/kaspersky/kess. If you specify a different directory, make sure that the account under which Kaspersky Embedded Systems Security is running has read/write permissions for this directory. Root privileges are required to access the default trace files directory.
Maximum number of trace files	In this field, you can enter the maximum number of application trace files. Default value: 10.
Maximum trace file size (MB)	In this field, you can enter the maximum size of an application trace file (in megabytes). Default value: 500.

To apply trace settings, you must restart the application.

Editing trace settings in the Administration Console

In the Administration Console, you can manage application trace settings in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls** → **Application settings**).

Under **Trace and dump settings**, click **Configure** to open a window in which you can edit the trace settings (see the table below).

Trace settings

Setting	Description
Path to the trace file directory	Input field for the path to the directory where the trace files are stored. Default value: /var/log/kaspersky/kess. If you specify a different directory, make sure that the account under which Kaspersky Embedded Systems Security is running has read/write permissions for this directory. Root privileges are required to access the default trace files directory.

Setting	Description
Maximum trace file size (MB)	In this field, you can enter the maximum size of an application trace file (in megabytes). Default value: 500.
Maximum number of trace files	In this field, you can enter the maximum number of application trace files. Default value: 10.

To apply trace settings, you must restart the application.

Editing trace settings in the command line

In the command line, you can configure application tracing settings using the `TraceLevel`, `TraceFolder`, `TraceMaxFileCount` and `TraceMaxFileSize` settings in general application settings.

The `TraceLevel` setting lets you enable or disable the creation of application trace files and specify the level of detail in trace files. This setting can take the following values:

- `Detailed` – Generate a detailed trace file.
- `MediumDetailed` – Generate a trace file that contains informational messages and error messages.
- `NotDetailed` – Generate a trace file that contains error messages.
- `None` (default value) — Do not generate a trace file.

The `TraceFolder` settings lets you specify the directory where application trace files are stored. Default value: /var/log/kaspersky/kess. If you specify a different directory, make sure that the account under which Kaspersky Embedded Systems Security is running has only read/write permissions for this directory (rwx------- 700 permissions). Root privileges are required to access the default trace files directory.

The `TraceMaxFileCount` setting lets you specify the maximum number of application trace files. The setting can take values from 1 to 10000. Default value: 10.

The `TraceMaxFileSize` setting lets you specify the maximum size of an application trace file (in megabytes). The setting can take values from 1 to 1000. Default value: 500.

You can change the values of the settings with the help of command line switches or a configuration file that contains all general application settings.

After changing the values of the `TraceFolder`, `TraceMaxFileCount`, or `TraceMaxFileSize` settings, you need to restart the application.

Application management plug-in trace files

Management plug-in trace files are not sent to Kaspersky automatically.

Trace files are saved in a human-readable format. It is recommended to protect information from unauthorized access before sending it to Kaspersky.

MMC administration plug-in trace files

If you use the Administration Console to manage Kaspersky Embedded Systems Security, information about events that occur while the administration MMC plug-in is running can be saved to the Kaspersky Embedded Systems Security MMC plug-in trace file on the device where the Administration Server is installed. The file name contains the application version number, file creation date and time, and process identifier (PID). This file contains information about the events that occur during MMC plug-in operation, in particular, about the operation of policies and tasks.

MMC plug-in trace files are not generated by default. You can use registry keys to create the MMC plug-in trace file. Contact Technical Support representatives for detailed information on how to create trace files.

All created trace files of the MMC plug-in are located in the folder specified by the user during registry key configuration.

Management web plug-in trace files

If you use the Web Console to manage Kaspersky Embedded Systems Security, information about events that occur while the administration web plug-in is running can be written to the web plug-in trace files.

Trace files for the web plug-in are created automatically if logging of Web Console activities is enabled in Web Console Installation Wizard (for more details, refer to the Kaspersky Security Center Help).

Trace files of the web plug-in are stored in the Web Console installation folder in the "logs" subfolder.

Contents of management plug-in trace files

Trace files contain the following general data:

- Event time.
- Number of the thread of execution.
- Application component that caused the event.

- Degree of event severity (informational event, warning, critical event, error).
- A description of the event involving command execution by a component of the application and the result of execution of this command.

In addition to general data, trace files may contain the following information:

- Personal data, including the last name, first name, and middle name, if such data is part of the path to files.
- The name of the account used to log in to the operating system if the user account name is part of a file name.

About dump files

A *dump file* contains all information about the working memory of Kaspersky Embedded Systems Security processes at the time of dump creation.

Dump files may contain personal data. It is recommended to protect information from unauthorized access before sending it to Kaspersky.

No dump files are generated by default. You can enable or disable dumping in case of application failures.

If you have enabled dumping, by default, dump files are stored in the following directories:

- /var/opt/kaspersky/kess/common/dumps (dump files of the kess process)
- /var/opt/kaspersky/kess/common/dumps-user (dump files of user processes)

Root privileges are required to access the default dump file directory. You can specify a different directory for storing the dump files of the kess process. If you specify a different directory, make sure that only the root has read and write permissions to this directory (`700` or `rwx--------` permissions).

To store dump files, the application needs disk space in addition to the disk space listed in minimum hardware requirements.

Dump files are stored on the computer as long as the application is in use, and are deleted permanently when the application is removed. Dump files are not sent to Kaspersky automatically.

Managing dump file settings

If you are managing Kaspersky Embedded Systems Security using Kaspersky Security Center, you can manage dump settings of the application in the Kaspersky Embedded Systems Security policy settings using the Web Console or the Administration Console.

If you use the command line to manage the application, you can enable or disable dumping via the kess.ini configuration file.

The maximum number of dump files is limited.

Depending on the operating system settings, user dump files may not be created. Make sure that the system kernel is configured using `sysctl kernel.yama.ptrace_scope=0`.

Managing dump file settings in the Web Console

In the Web Console, you can configure application dump file settings in the policy properties (**Application settings** → **General Settings** → **Technical Support**, **Debug information** section).

Dump file settings

Setting	Description
Dumping enabled / disabled	If dumpis is enabled, Kaspersky Embedded Systems Security creates a dump file if the application crashes. This check box is cleared by default.
Code dump path	Input field for the path to the directory where the dump files are stored. The input field is limited to 128 characters. Default value: /var/opt/kaspersky/kess/common/dumps.

You must restart the application to apply the dump file settings.

Managing dump file settings in the Administration Console

In the Administration Console, you can manage dumping settings for the application in the policy properties (Managing policies in the Administration Console, Creating a policy using the Administration Console) (**Security Controls** → **Application settings**).

Under **Trace and dump settings**, click **Configure** to open a window in which you can edit the dumping settings (see the table below).

Dump file settings

Setting	Description
Create a dump file if the application crashes	This check box enables or disables the creation of a dump file when the application crashes. This check box is cleared by default.
Path to the dump file directory	Input field for the path to the directory where the dump files are stored. The input field is limited to 128 characters. Default value: /var/opt/kaspersky/kess/common/dumps.

You must restart the application to apply the dump file settings.

Managing dump file settings on the command line

To enable or disable dumping via the kess.ini configuration file, do as follows:

1. Stop Kaspersky Embedded Systems Security.
2. Open the /var/opt/kaspersky/kess/common/kess.ini file for editing.
3. Under **[General]**, set the parameter value:
   - `CoreDumps=yes`: enable dumping in case of a failure.
   - `CoreDumps=no`: disable dumping.
4. If you want to change the default directory where dump files are saved, specify the path to the directory in the `CoreDumpsPath` option.
5. Start Kaspersky Embedded Systems Security.

Remote device diagnostics using Kaspersky Security Center

In Kaspersky Security Center you can perform remote diagnostics of client devices. The remote diagnostics procedure lets you remotely run the following operations:

- Enable or disable tracing.
- Change the trace level.
- Download trace files.
- Download a remote application installation log.
- Download system event (syslog) logs.
- Start, stop, and restart applications.

Remote diagnostics in the Web Console

If you use the Web Console to manage Kaspersky Embedded Systems Security, remote diagnostics of a client device is done in the remote diagnostics window.

To open the remote device diagnostics window for a device:

1. In the main window of the Web Console, select **Assets (Devices)** → **Managed devices**.

   The list of managed devices opens.

2. Select a device that you want to diagnose remotely and click the <device name> link.

   The device properties window opens.

3. On the **Advanced** tab, select the **Remote diagnostics** section.

In the device remote diagnostics window, you can view the remote installation log.

To view the remote installation log on a device, do as follows:

1. Open the remote device diagnostics window.
2. On the **Event logs** tab, under **Trace files** block, click **Remote installation logs**.

   The **Device trace event logs** window opens.

For more information about the remote diagnostics, see the Kaspersky Security Center Help.

Remote diagnostics using the Administration Console

If you use the Administration Console to manage Kaspersky Embedded Systems Security, remote diagnostics is done using the special Kaspersky Security Center remote diagnostics utility automatically installed on the device together with the Administration Console.

To open the main window of the remote diagnostics utility, do as follows:

1. In the Administration Console tree, in the **Managed devices** folder, select the administration group containing the necessary device.
2. In the workspace, select the **Devices** tab.
3. In the list of managed devices, select the device to which you want to connect the remote diagnostics utility, and select **External tools** → **Remote diagnostics** in the device context menu.

   The main window of the **Kaspersky Security Center remote diagnostics utility** opens.

You can use the remote device diagnostics utility to view the remote installation log.

To view the remote installation log on a device, do as follows:

1. Open the main window of the remote diagnostics utility.
2. Configure the options for connecting the utility to the device if needed. In the main window of the remote diagnostics utility, click the **Log in** button.
3. In the window that opens, in the objects tree, select the **Remote installation logs** folder.

For more information about the remote diagnostics utility, refer to Kaspersky Security Center Help section.

Appendices

This section provides information that complements the primary Help text.

Appendix 1. Commands for managing the application

On the command line, Kaspersky Embedded Systems Security can be managed using Kaspersky Embedded Systems Security management commands.

You can view the help on management commands by running:

`kess-control --help <command group prefix>`

Where <`command group prefix`> accepts the following values:

- -A: commands for managing Application Control
- -B: commands for managing Backup
- -D: commands for managing Device Control
- -E: commands for managing application events (Viewing events on the command line, Viewing events in Kaspersky Security Center)
- -F: commands for managing firewall
- -H: commands for managing blocked devices
- -J: commands for managing the obtaining of system performance metrics
- -L: commands for managing license keys
- -N: commands for managing network traffic scan settings
- -Q: commands for managing the Quarantine
- -R: commands for managing device network isolation
- -S: statistics commands
- -T: commands for managing application tasks and settings
- -U: commands for managing users and roles
- -W: event output commands (Viewing events on the command line, Viewing events in Kaspersky Security Center)

Commands for managing application tasks and settings

The -T prefix indicates that the command belongs to the group of commands for managing application settings and tasks.

kess-control --export-settings

This command outputs all application settings to the console or exports to a configuration file. These settings include encrypted connections scan settings, general application settings, and task settings.

Command syntax

```
kess-control [-T] --export-settings [--file <path to configuration file>] [--json]
```

Arguments and keys

`--file <configuration file path>` is the full path to the configuration file where the application settings will be saved.

`--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

kess-control --import-settings

This command imports all application settings from a configuration file, including encrypted connections scan settings, general application settings, and task settings.

Command syntax

```
kess-control [-T] --import-settings --file <path to configuration file> [--json]
```

Arguments and keys

`--file <configuration file path>` is the full path to the configuration file from which the settings will be imported into the application.

`--json` is specified to import the settings from the configuration file in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

kess-control --update-application

This command installs a downloaded application module update.

Command syntax

```
kess-control [-T] --update-application
```

Commands for managing general application settings

The kess-control --get-app-settings command

The command outputs the current values of the general application settings to the console or a configuration file.

Command syntax

```
kess-control [-T] --get-app-settings [--file <path to configuration file>] [--json]
```

Arguments and keys

`--file <configuration file path>` is the path to the configuration file where the application general settings will be written. If you do not specify the `--file` option, settings will be output to the console.

If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, no configuration file will be generated.

`--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

The kess-control --set-app-settings command

This command configures the general application settings via command options or by importing settings from a configuration file.

Command syntax

Define settings via command options:

```
kess-control [-T] --set-app-settings <setting name>=<setting value> [<setting name>=
<setting value>]
```

Define settings via a configuration file:

```
kess-control [-T] --set-app-settings --file <path to configuration file> [--json]
```

Arguments and keys

`<setting name>=<setting value>` is the name and value of one of the general application settings.

`--file <configuration file path>` is the full path to the configuration file from which the settings will be imported into the application.

`--json` is specified to import the settings from the configuration file into the application in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

Commands for managing task settings

kess-control --get-settings

This command outputs the current settings for a specified task to the console or a configuration file.

Command syntax

```
kess-control [-T] --get-settings <task ID/name> [--file <path to configuration file>] [--json]
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

`--file <configuration file path>` is the path to the configuration file where the task settings will be written. If you do not specify the `--file` option, settings will be output to the console.

If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, no configuration file will be generated.

`--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

kess-control --set-settings

This command defines the settings for a specified task via command options or by importing settings from a configuration file.

Command syntax

Define settings via command options:

```
kess-control [-T] --set-settings <task ID/name> <setting name>=<setting value> [<setting name>=<setting value>] [--add-path <path>] [--del-path <path>] [--add-exclusion <path>] [--del-exclusion <path>]
```

Define settings via a configuration file:

```
kess-control [-T] --set-settings <task ID/name> --file <path to configuration file> [--json]
```

Arguments and keys

< task ID/name > is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

< setting name >=< setting value > is the name and value of one of the task settings.

--add-path < path > adds a path to a directory with objects to be scanned.

--del-path < path > removes a path to a directory with objects to be scanned.

--add-exclusion < path > adds a path to a directory with objects to be excluded from scanning.

--del-exclusion < path > removes a path to a directory with objects to be excluded from scanning.

--file < configuration file path > is the full path to the configuration file from which the task settings will be imported.

--json is specified to import the settings from the configuration file in JSON format. If the --json option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

kess-control --set-to-default

The command restores the default settings for the specified task.

Command syntax

```
kess-control [-T] --set-settings < task ID/name > --set-to-default
```

Arguments and keys

< task ID/name > is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

The kess-control --get-schedule command

The command outputs the current schedule of the specified task to the console or a configuration file.

Command syntax

```
kess-control [-T] --get-schedule < task ID/name > [--file < path to configuration file >] [--json]
```

Arguments and keys

< task ID/name > is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

--file < configuration file path > is the path to the configuration file where the task schedule settings will be written. If you do not specify the --file option, settings will be output to the console.

If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, no configuration file will be generated.

`--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

The kess-control --set-schedule command

The command defines a schedule for the specified task via command options or by importing settings from a configuration file.

Command syntax

Define settings via command options:

```
kess-control [-T] --set-schedule <task ID/name> <setting name>=<setting value>
[<setting name>=<setting value>]
```

Define settings via a configuration file:

```
kess-control [-T] --set-schedule <task ID/name> --file <path to configuration file> [--
json]
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

`<setting name>=<setting value>` is the name and value of one of the task schedule settings.

`--file <configuration file path>` is the full path to the configuration file from which the task schedule settings will be imported.

`--json` is specified to import the settings from the configuration file in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

Commands for managing tasks

kess-control --get-task-list

This command outputs a list of existing tasks.

Command syntax

```
kess-control [-T] --get-task-list [--json]
```

Arguments and keys

`--json` is specified to output the settings in JSON format.

kess-control --get-task-state

This command outputs the status of the specified task.

Command syntax

```
kess-control [-T] --get-task-state <task ID/name> [--json]
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

`--json` is specified to output the settings in JSON format.

kess-control --create-task

This command creates a task of the specified type with the default settings or settings specified in a configuration file.

Command syntax

Create a task with the default settings:

```
kess-control [-T] --create-task <task name> --type <task type>
```

Create a task with the settings from a configuration file:

```
kess-control [-T] --create-task <task name> --type <task type> [--file <path to configuration file>] [--json]
```

Arguments and keys

`<task name>` is the name that you specify for the new task.

`<task type>` is the designation of the type of the task you are creating.

`--file <path to configuration file>` is the full path to the configuration file with the settings that will be used for creating the task.

`--json` is specified to import the settings from the configuration file in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

kess-control --delete-task

This command deletes a task.

Command syntax

```
kess-control [-T] --delete-task <task ID/name>
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

kess-control --start-task

This command starts a task.

Command syntax

```
kess-control [-T] --start-task <task ID/name> [-W] [--progress]
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

`[-W]`: enable current events output.

`[--progress]`: display task progress.

kess-control --stop-task

This command stops a task.

Command syntax

```
kess-control [-T] --stop-task <task ID/name> [-W]
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

`[-W]`: enable current events output.

kess-control --suspend-task

This command pauses a task.

Command syntax

```
kess-control [-T] --suspend-task <task ID/name>
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

kess-control --resume-task

This command resumes a task.

Command syntax

```
kess-control [-T] --resume-task <task ID/name>
```

Arguments and keys

`<task ID/name>` is the ID assigned to the task at the time of its creation, or the name of the task in the command line.

kess-control --scan-file

This command creates and runs a custom scan task.

Command syntax

```
kess-control [-T] --scan-file <path> [--action <action>]
```

Arguments and keys

`<path>` is the path to the file or directory that you want to scan. You can specify multiple paths by separating them with a space.

`--action <action>` is the action to be performed by the application on the infected objects. If you do not specify the `--action` option, the application performs the recommended action.

Commands for managing network traffic scan settings

The -N prefix indicates that the command belongs to the group of commands for managing network traffic scan settings.

kess-control -N --query

The command outputs lists of exclusions from encrypted connections scanning:

- a list of exclusions added by the user;
- a list of exclusions added by the application;
- list of exclusions received from the application databases.

Command syntax

```
kess-control -N --query user

kess-control -N --query auto

kess-control -N --query kl
```

kess-control --clear-web-auto-excluded

This command clears the list of domains that the application has automatically excluded from scanning.

Command syntax

```
kess-control -N --clear-web-auto-excluded
```

kess-control --get-net-settings

This command outputs the current network traffic scan settings to the console or a configuration file.

Command syntax

```
kess-control [-N] --get-net-settings [--file <path to configuration file>] [--json]
```

Arguments and keys

`--file <configuration file path>` is the path to the configuration file to which network traffic scan settings will be written. If you do not specify the `--file` option, settings will be output to the console.

If you specify the name of a file without its path, the file will be created in the current directory. If a file already exists in the specified path, it will be overwritten. If the specified directory does not exist, no configuration file will be generated.

`--json` is specified to output the settings in JSON format. If the `--json` option is not specified, the settings are output in the INI format.

kess-control --set-net-settings

This command configures the network traffic scan settings with command options or by importing settings from a configuration file.

Command syntax

Define settings via command options:

```
kess-control [-N] --set-net-settings <setting name>=<setting value> [<setting name>=<setting value>]
```

Define settings via a configuration file:

```
kess-control [-N] --set-net-settings --file <path to configuration file> [--json]
```

Arguments and keys

`<setting name>=<setting value>` is the name and value of one of the network traffic scan settings.

`--file <configuration file path>` is the full path to the configuration file from which the network traffic scan settings will be imported.

`--json` is specified to import the settings from the configuration file into the application in JSON format. If the `--json` option is not specified, the application attempts to import from an INI file. If the import fails, an error is displayed.

kess-control --list-certificates

This command outputs a list of trusted root certificates.

Command syntax

```
kess-control [-N] --list-certificates
```

kess-control --add-certificate

This command adds a certificate to the list of trusted root certificates.

Command syntax

```
kess-control [-N] --add-certificate <path to certificate>
```

Arguments and keys

`<path to certificate>` is the path to the certificate file that you want to add (PEM or DER format).

kess-control --remove-certificate

This command removes a certificate from the list of trusted root certificates.

Command syntax

```
kess-control [-N] --remove-certificate <certificate subject>
```

kess-control --add-bypass-endpoints

This command lets you add traffic interception exclusion.

Command syntax

```
kess-control [-N] --add-bypass-endpoints --direction <in|out> --remote-ip <remote IP address> --dst-port <destination port>
```

Arguments and keys

`<in|out>` is the direction of traffic in the excluded connection, outbound or inbound.

`<remote IP address>` is the IPv4 or IPv6 address of the remote device. If you exclude inbound traffic from interception, traffic coming from the specified IP address is excluded. If you exclude outbound traffic from interception, traffic going to the specified IP address is excluded.

`<destination port>` is a port on the protected device (exclusion for inbound traffic) or a port on the remote device (exclusion for outbound traffic). Traffic to this port is excluded from interception.

kess-control --list-bypass-endpoints

This command lets you view the list of configured traffic interception exclusions.

Command syntax

```
kess-control [-N] --list-bypass-endpoints
```

kess-control --remove-bypass-endpoints

This command lets you remove a traffic interception exclusion.

Command syntax

```
kess-control [-N] --remove-bypass-endpoints --direction <in|out> --remote-ip <remote IP address> --dst-port <destination port>
```

Arguments and keys

`<in|out>` is the direction of traffic in the excluded connection, outbound or inbound.

`< remote IP address >` is the IPv4 or IPv6 address of the remote device whose traffic is excluded from interception (depending on the direction of traffic).

`< destination port >` is a port on the protected device (exclusion for inbound traffic) or a port on the remote device (exclusion for outbound traffic).

Statistics commands

The -S prefix indicates that the command belongs to the statistics command group.

kess-control --app-info

This command outputs information about the application.

Command syntax

```
kess-control [-S] --app-info [--json]
```

Arguments and keys

`--json` is specified to output the settings in JSON format.

kess-control --supported-tech-info

This command displays a list of application functions, information about their status (used or not used), and the Linux technologies that these application features are implemented with.

Command syntax

```
kess-control [-S] --supported-tech-info [--json]
```

Arguments and keys

`--json`: output features in JSON format.

kess-control --get-statistic

The command allows you to display statistics about the operation of the application and the list of mount points found on the device.

Command syntax

```
kess-control [-S] --get-statistic [--files] [--processes] [--mountpoints]
```

Arguments and keys

`[--files]` : statistics of files most frequently scanned by the File Threat Protection component, and the number of times the component accesses these files.

`[--processes]` : statistics of process starts most frequently detected by Kaspersky Embedded Systems Security, and the number of such starts.

> Excluding processes from scanning by the Behavior Detection component does not remove these processes from statistics, but it improves the performance of the Kaspersky Embedded Systems Security application.

`[--mountpoints]` : list of mount points. In the output, paths to mount points are sorted as follows:

- Static, system, service and other mount points
- Mounted snap packages and mount points in their namespaces

You can specify one or more options in any combination or no options at all. If you do not specify any options, the application displays three lists: statistics on the most frequently scanned files, statistics on the most frequently started processes, and the list of mount points found on the device.

kess-control --omsinfo

This command creates a JSON file for integration with Microsoft Operations Management Suite.

Command syntax

```
kess-control [-S] --omsinfo --file <file path>
```

Commands for displaying events

kess-control -W

This command enables the display of current application events. The command returns the name of the event and additional information about the event. You can use the command to display all current application events or only events associated with a currently running task.

Command syntax

```
kess-control -W [--query "<filter conditions>"]
```

Arguments and keys

`<filter conditions>` specifies one or more logical expressions in the `<field> <comparison operation> '<value>'` format combined using the `and` logical operator to output certain current events.

Commands for managing application events

The -E prefix indicates that the command belongs to commands for managing application events (Viewing events on the command line, Viewing events in Kaspersky Security Center).

kess-control -E

This command outputs information about all events in the application event log. You can use the `less` command to navigate through the list of displayed events.

Command syntax

```
kess-control -E
```

kess-control -E --query

This command outputs information about events from the application event log. You can use the `less` command to navigate through the list of displayed events. You can use a filter to output specific events or output a list of events to a file.

Command syntax

```
kess-control -E --query "<filter conditions>" [--db <database file>] [-n <number>] [--file <file path>] [--json] [--reverse]
```

Arguments and keys

`<database file>` is the full path to the event log database file to output events from. By default, the application saves information about events to the /var/opt/kaspersky/kess/private/storage/events.db database. The location of the database is determined by the `EventsStoragePath` general application setting.

`<filter conditions>` specifies one or more logical expressions in the `<field> <comparison operation> '<value>'` format combined using the `and` logical operator to restrict query results.

`<number>` is the number of the latest events of the selection (records counting from the end of the selection) to be displayed.

`--file <file path>` is the full path to the file to output events to. If you specify the name of a file without specifying its path, the file will be created in the current directory. If a file with the specified name already exists in the specified path, it will be overwritten. If the specified directory cannot be found on the disk, file will not be created.

If you do not specify the `--file` option, the list of events will be output to the console.

`--json`: output events in JSON format.

`--reverse`: display events in reverse order (from the newest event at the top to the oldest at the bottom).

Commands for managing license keys

The -L prefix indicates that the command belongs to the group of commands used to manage license keys.

kess-control --add-active-key

The command lets you add an active license key to the application using a key file or activation code.

You can use this command to add main license keys of the application as well as MDR license keys. You do not need to specify the type of the key in the command.

Command syntax

```
kess-control [-L] --add-active-key <path to the key file>

kess-control [-L] --add-active-key <activation code>
```

Arguments and keys

`<path to the key file>` is the path to the key file. If the key file is located in the current directory, it is sufficient to specify only the file name.

`<activation code>` is the activation code.

Example:
Add a key as an active key from the /home/test/00000001.key file:

```
kess-control --add-active-key /home/test/00000001.key
```

kess-control --add-reserve-key

The command lets you add a reserve license key to the application using a key file or an activation code.

You can use this command to add main license keys of the application as well as MDR license keys. You do not need to specify the type of the key in the command.

If an active key has not yet been added to the application on the device, the command fails.

Command syntax

```
kess-control [-L] --add-reserve-key <path to the key file>

kess-control [-L] --add-reserve-key <activation code>
```

Arguments and keys

< path to the key file > is the path to the key file. If the key file is located in the current directory, it is sufficient to specify only the file name.

< activation code > is the activation code.

Example:
Add a reserve key using the /home/test/00000002.key file:

```
kess-control --add-reserve-key /home/test/00000002.key
```

kess-control --remove-active-key

This command lets you remove an active license key.

Command syntax

```
kess-control [-L] --remove-active-key [--mdr]
```

Arguments and keys

`--mdr` – remove the active MDR license key. If you do not specify the `--mdr` option, the main active key is removed.

kess-control --remove-reserve-key

This command lets you remove a reserve license key.

Command syntax

```
kess-control [-L] --remove-reserve-key [--mdr]
```

Arguments and keys

`--mdr` – remove the reserve MDR license key. If you do not specify the `--mdr` option, the main reserve key is removed.

kess-control -L --query

The `-L --query` command outputs information about the license that was used for activating the application and license keys currently in use.

Command syntax

```
kess-control -L --query [--json]
```

Arguments and keys

`--json`: output data in JSON format.

Commands for Firewall Management

The -F prefix indicates that the command belongs to the firewall management commands.

kess-control --add-rule

This command adds a new network packet rule.

Command syntax

```
kess-control [-F] --add-rule [--name <rule name>] [--action <action>] [--protocol
<protocol>] [--direction <direction>] [--remote <remote address>[:<port range>]] [--
local <local address>[:<port range>]] [--at <index>]
```

Arguments and keys

`--name <rule name>` is the name of the network packet rule.

`--action <action>` is the action to be performed on connections specified in network packet rule.

`--protocol <protocol>` is the type of communication protocol for which you want to monitor network activity.

`--direction <direction>` is the direction of the monitored network activity.

`--remote <remote address>[:<port range>]` is the network address of the remote device.

`--local <local address>[:<port range>]` is the network address of the Kaspersky Embedded Systems Security device.

`--at <index>` is the number of the rule in the list of network packet rules. If the `--at` option is not specified or its value is larger than the number of rules in the list, the new rule is added to the end of the list.

Parameters that you do not specify values for in the command are set to their default values.

kess-control --del-rule

This command deletes the network packet rule with the specified name or index in the list of rules.

Command syntax

```
kess-control [-F] --del-rule --name <rule name>
```

```
kess-control [-F] --del-rule --index <index>
```

Arguments and keys

`--name <rule name>` is the name of the network packet rule.

`--index <index>` is the number of the rule in the list of network packet rules.

kess-control --move-rule

This command changes the execution priority of a network packet rule.

Command syntax

```
kess-control [-F] --move-rule --name <rule name> --at <index>

kess-control [-F] --move-rule --index <index> --at <index>
```

Arguments and keys

`--name <rule name>` is the name of the network packet rule.

`--index <index>` is the current number of the rule in the list of network packet rules.

`--at <index>` is the new number of the rule in the list of network packet rules.

kess-control --add-zone

This command adds an address to a network zone.

Command syntax

```
kess-control [-F] --add-zone --zone <zone> --address <address>
```

Arguments and keys

`--zone <zone>` is the predefined name of the network zone.

`--address <address>` is the network address or subnet.

kess-control --del-zone

This command removes an address from a network zone.

Command syntax

```
kess-control [-F] --del-zone --zone <zone> --address <address>

kess-control [-F] --del-zone --zone <zone> --index <address index>
```

Arguments and keys

`--zone <zone>` is the predefined name of the network zone.

`--address <address>` is the network address or subnet.

`--index <address index>` is the number of the address in the network zone.

kess-control -F --query

This command displays firewall rules created using Kaspersky Embedded Systems Security.

Command syntax

```
kess-control -F --query
```

Commands used to manage blocked devices

The -H prefix indicates that the command belongs to the group of commands for managing devices blocked by Anti-Cryptor and Network Threat Protection.

kess-control --get-blocked-hosts

The command allows you to output the list of blocked devices to the console.

Command syntax

```
kess-control [-H] --get-blocked-hosts
```

kess-control --allow-hosts

The command allows you to unblock blocked devices.

Command syntax

```
kess-control [-H] --allow-hosts <address>
```

Arguments and keys

`<address>` is an IP address of a device or subnet (IPv4/IPv6, including addresses in short form). You can specify multiple IP addresses of devices or subnets by separating them with a space.

Commands for managing Device Control

The D prefix indicates that the command belongs to the group of commands to manage Device Control.

kess-control --get-device-list

This command outputs the list of devices installed on a host or connected to it, or exports a list of connected devices to a JSON file.

Command syntax

```
kess-control [-D] --get-device-list [--all] [--json] [--export <file>]
```

Arguments and keys

`--all` lists all connected devices.

`--json` lists connected devices in JSON format.

`--export <file>` exports the list of connected devices to a file, formatted as a list of trusted devices. You need to specify the full path to the JSON file.

kess-control --trusted-devices

The command lets you view the list of trusted devices, add and remove devices from the list, and import the list of trusted devices from a file.

Command syntax

```
kess-control [-D] --trusted-devices [--list] [--add <id>] [--delete <id>] [--merge|--add|--replace --file <file>]
```

Arguments and keys

`--list` – display the list of trusted devices.

`--add <ID>` adds a device to the list of trusted devices by device ID or device ID mask.

`--delete <ID>` removes a device from the list of trusted devices by device ID or device ID mask.

`--merge` merges the lists of trusted devices. If the current list and the imported list contain entries for a device with the same ID, the existing entries are updated. For devices that are not in the current list, entries are added.

`--add` adds the current list of trusted devices when importing. Only entries for devices not previously on the list of devices are added to the list.

`--replace` replaces the current list of trusted devices. The existing list of trusted devices is replaced with the imported list.

`--file <file>` imports a list of trusted devices from the specified file. You need to specify the full path to the JSON file.

kess-control --request-temporary-device-access

This command lets you create an access request file to request temporary access to a blocked device.

Command syntax

```
kess-control [-D] --request-temporary-device-access --device-id <device ID> --access-duration <access duration> --path <path to access request file>
```

Arguments and keys

`--device-id <device ID>` specifies the ID of the blocked device to which you want to gain access.

`--access-duration <access duration>` specifies the time period during which you want to be able to gain access to the device (in hours). Possible values: 1 to 100

`--path <path to the access request file>` specifies the path to the directory in which you want to save the access request file with the .akey extension.

kess-control --upload-temporary-device-access-key

This command allows you to upload an access key to gain temporary access to a blocked device.

Command syntax

```
kess-control [-D] --upload-temporary-device-access-key --path <path to the file with the access key>
```

Arguments and keys

`--path <path to the file with the access key>` is the path to the file containing the access key to the device (file with the .acode extension).

Commands for managing Application Control

The -A prefix indicates that the command belongs to the group of commands to manage Application Control.

kess-control --get-app-list

The command outputs a list of applications found on a client device by the Inventory task.

Command syntax

```
kess-control [-A] --get-app-list [--json]
```

Arguments and keys

`--json`: output data in JSON format.

kess-control --get-categories

This command outputs a list of created application control categories.

Command syntax

```
kess-control [-A] --get-categories [--names <name of category 1> <name of category 2>
... <name of category N>] [--file <path to configuration file>] [--json]
```

Arguments and keys

`<name of category 1> <name of category 2> ... <name of category N>` are names of categories for which you want to view information. If you want to view information about several categories, specify the names of the categories, separated by a space.

`--file <path to configuration file>` is the full path to the JSON configuration file to which the settings will be written.

`--json`: output data in JSON format.

kess-control --set-categories

This command lets you create or edit the list of created Application Control categories.

Command syntax

```
kess-control [-A] --set-categories [--names <name of category 1> <name of category 2>
... <name of category N>] --file <path to configuration file>
```

Arguments and keys

`<name of category 1> <name of category 2> ... <name of category N>` are names of categories for which you want to change information. If you want to change information about several categories, specify the names of the categories, separated by a space. If you do not specify a category name, the category will be removed from the list.

`--file <path to configuration file>` is the path to the configuration file with the category settings.

kess-control --get-settings 21

This command outputs a list of created application control rules.

Command syntax

```
kess-control --get-settings 21 [--file <path to configuration file>] [--json]
```

Arguments and keys

`--file <path to configuration file>` is the full path to the configuration file to which the settings will be written.

`--json`: output data in JSON format.

kess-control --set-settings 21

This command lets you edit the list of created application categories and Application Control rules.

Command syntax

```
kess-control --set-settings 21 [--file <path to configuration file>] [--json]
```

Arguments and keys

`--file <path to configuration file>` is the full path to the configuration file from which the settings will be imported.

`--json` – import data from a JSON file.

kess-control --set-to-default 21

This command lets you delete a list of application categories and Application Control rules.

Command syntax

```
kess-control --set-settings 21 --set-to-default
```

kess-control ---add-app-control-trust-certificate

This command adds a certificate to Application Control's list of trusted certificates.

Command syntax

```
kess-control [-A] --add-app-control-trust-certificate <path to certificate>
```

Arguments and keys

<path to certificate> is the path to the certificate file that you want to add (PEM or DER format).

kess-control --remove-app-control-trust-certificate

This command removes a certificate from Application Control's list of trusted certificates.

Command syntax

```
kess-control [-A] --remove-app-control-trust-certificate <serial number of certificate>
```

kess-control --query-app-control-trust-certificates

This command outputs a list of Application Control's trusted certificates.

Command syntax

```
kess-control [-A] --query-app-control-trust-certificates
```

Commands for managing Backup

The -B prefix indicates that the command belongs to the group of commands used to manage the Backup storage.

kess-control -B --mass-remove

The command deletes some or all objects from Backup.

Command syntax

Delete all objects:

```
kess-control -B --mass-remove
```

Delete objects that match the filter conditions:

```
kess-control -B --mass-remove --query "<filter conditions>"
```

Arguments and keys

<filter conditions> specifies one or more logical expressions in the <field> <comparison operation> '<value>' format combined using the and logical operator to restrict query results.

kess-control -B --query

This command outputs information about Backup objects.

Command syntax

Output information about all objects in Backup:

```
kess-control -B --query [-n <number>] [--json] [--reverse]
```

Output information about Backup objects that match the filter conditions:

```
kess-control -B --query ["<filter conditions>"] [-n <number>] [--json] [--reverse]
```

Arguments and keys

<filter conditions> specifies one or more logical expressions in the <field> <comparison operation> '<value>' format combined using the and logical operator to restrict query results. If you do not specify any filter conditions, the application will display the details of all objects in Backup.

<number> is the number of the most recent objects to display. If you do not specify the -n switch, the last 30 objects will be displayed. To display all objects, specify 0.

--json: output data in JSON format.

--reverse – output objects in reverse order (from the newest object at the top to the oldest at the bottom).

kess-control -B --restore

This command restores an object from Backup.

Command syntax

```
kess-control -B --restore <object ID> [--file <file path>]
```

Arguments and keys

<object ID> is the ID of the Backup object.

--file <file path> is the new name of the file and the path to the directory to save it to. If you do not specify the --file option, the object will be restored with its original name and to its original location.

Commands for managing the Quarantine

The -Q prefix indicates that the command belongs to the group of commands used to manage the Quarantine.

kess-control -Q --mass-remove

The command deletes some or all files from Quarantine.

Command syntax

Delete all files:

```
kess-control -Q --mass-remove
```

Delete files that match the filter conditions:

```
kess-control -Q --mass-remove --query "< filter conditions >"
```

Arguments and keys

`< filter conditions >` specifies one or more logical expressions in the `< field > < comparison operation > '< value >'` format combined using the `and` logical operator to restrict query results.

kess-control --put

The command allows to quarantine a file.

Command syntax

```
kess-control [-Q] --put < file path > [--md5] [--sha256] [--save-original-file]
```

Arguments and keys

`< file path >` is the path to the file that you want to quarantine.

`--md5` is the MD5 hash of the file that you want to quarantine.

`--sha256` is the SHA256 hash of the file that you want to quarantine.

`--save-original-file` keeps the original file. If you do not specify this option, the original file is deleted.

kess-control -Q --query

This command outputs information about quarantined files.

Command syntax

Output information about all quarantined files:

```
kess-control -Q --query [-n <number>] [--json]
```

Display information about quarantined files that match filter conditions:

```
kess-control -Q --query ["<filter conditions>"] [-n <number>]
```

Arguments and keys

`<filter conditions>` specifies one or more logical expressions in the `<field> <comparison operation> '<value>'` format combined using the `and` logical operator to restrict query results. If you do not specify any filter conditions, the application will display the details of all quarantined files.

`<number>` is the count of the most recent quarantined files that you want to display. If you do not specify the `-n` option, the last 30 files are displayed. To display all files, specify 0.

`--json`: output data in JSON format.

kess-control -Q --restore

The command lets you restore a file from quarantine.

Command syntax

```
kess-control -Q --restore <file ID> [--file <file path>]
```

Arguments and keys

`<file ID>`: the ID of the quarantined file.

`--file <file path>` is the new name of the file and the path to the directory to save it to. If you do not specify the `--file` option, the file will be restored with its original name and to its original location. If the directory is deleted or the user does not have access rights to it, the application places the file in the /var/opt/kaspersky/kess/common/restored/ directory. You can manually move the file from this directory to the directory of your choice.

Commands for managing device network isolation

The -R prefix indicates that the command belongs to the group of commands to manage the Network isolation of the device. Network isolation of devices is applied as a threat response action on threat detection when integrated with the Kaspersky Managed Detection and Response solution.

kess-control --isolation-stat

This command outputs the current state of network isolation to the console: enabled or disabled.

Command syntax

```
kess-control [-R] --isolation-stat
```

kess-control --isolation-off

This command lets you disable network isolation of a device.

Command syntax

```
kess-control [-R] --isolation-off
```

Commands for managing users and roles

The -U prefix indicates that the command belongs to the group of commands for managing users and roles.

kess-control --get-user-list

This command outputs a list of users and roles.

Command syntax

```
kess-control [-U] --get-user-list
```

kess-control --grant-role

This command assigns a role to a specific user.

Command syntax

```
kess-control [-U] --grant-role <role> <user>
```

kess-control --revoke-role

This command revokes a role from a specific user.

Command syntax

```
kess-control [-U] --revoke-role <role> <user>
```

Commands for managing system performance metrics

kess-control --export-metrics

This command allows to configure the obtaining of operating system performance metrics.

Command syntax

```
kess-control [-J] --export-metrics [--period <interval between exports in seconds>|--interactive]
```

Arguments and keys

`--period` enables periodic output of results.

`<interval in seconds between exports>` (in seconds) sets the output period.

`--interactive` enables interactive output (on the **Enter** key being pressed).

Appendix 2. Configuration files and default application settings

The following configuration files are used for managing Kaspersky Embedded Systems Security:

- Configuration files that contain the initial configuration settings of the application:
  - autoinstall.ini configuration file, used when installing the application via Kaspersky Security Center.
  - Configuration file used when installing the application via the command line.
- Preset configuration files generated automatically during the initial configuration of the application and containing the options set during the initial configuration. These settings are applied at run time.
- Configuration files that you can create with Kaspersky Embedded Systems Security management commands. These configuration files may contain task settings and other application settings. You can modify these files and import into the application to modify the corresponding options.

Rules for editing task configuration files

When editing a configuration file, adhere to the following rules:

- Specify all mandatory settings in the configuration file. You can specify individual task settings without a file using the command line.
- If a setting belongs to a certain section, specify it only in this section. You can specify the settings in any order within the one section.
- Enclose the names of sections in square brackets [ ].
- Specify the values of settings in the `< setting name >=< setting value >` format (spaces between the setting name and its value are not processed).

  Example:

  ```
  [ScanScope.item_0000]
  AreaDesc=Home
  AreaMask.item_0000=*doc
  Path=/home
  ```

  Space and tab characters are ignored before the first quotation mark and after the last quotation mark of a string value, and at the beginning and end of a string value that is not enclosed in quotation marks.
- If you need to specify several values for a setting, repeat the setting the same number of times as the number of values that you want to specify.

  Example:

  ```
  AreaMask.item_0000=*xml
  AreaMask.item_0001=*doc
  ```
- Be case-sensitive when entering values for the following types of settings:
  - Names (masks) of scanned objects and excluded objects.
  - Names (masks) of threats.

  The remaining setting values are not case-sensitive.
- Specify Boolean setting values as follows: `Yes / No`.
- Use quotation marks to enclose string values containing a space character (for example, names of files and directories and their paths, expressions containing the date and time in the format "YYYY-MM-DD HH:MM:SS").

  You can enter the remaining values with or without quotation marks.

  Example:

  ```
  AreaDesc=" Scanning of email databases "
  ```

  A single quotation mark in the beginning or end of a string is considered an error.

Preset configuration files

After the initial configuration, the application creates the following configuration files:

- /var/opt/kaspersky/kess/common/agreements.ini

  The agreements.ini configuration file contains settings related to the License Agreement, Privacy Policy, and Kaspersky Security Network Statement.

- /var/opt/kaspersky/kess/common/kess.ini

  The kess.ini configuration file contains the settings described in the following table.

If necessary, you can edit the values of the settings in these files.

The default values in these files should be changed only under the supervision of Technical Support specialists and in accordance with their instructions.

The kess.ini configuration file settings

Setting	Description	Values
The **[General]** section contains the following settings:		
`Locale`	The locale used for the localization of texts sent by Kaspersky Embedded Systems Security to Kaspersky Security Center (events, notifications, task results, etc.). The locale of the graphical interface and the application command line depends on the value of the `LANG` environment variable. If the locale that is not supported by Kaspersky Embedded Systems Security is specified as the value of the `LANG` environment variable, the graphical interface and the command line are displayed in English.	The locale in the format specified by RFC 3066. If the `Locale` setting is not specified, the operating system locale is used. If the application fails to determine the operating system localization language or the operating system localization is not supported, the default value will be used – `en_US.utf8`.
`PackageType`	Format of the installed application package. We do not recommend changing the value of this setting manually. The value of the setting is filled in automatically during initial application configuration.	`rpm` – an RPM package is installed. `deb` – a DEB package is installed.
`UseFanotify`	Using the fanotify technology to intercept file operations. We do not recommend changing the value of this setting manually. This setting is specified during the initial configuration of the application.	`true/yes` – the application uses the fanotify technology to intercept file operations. `false/no` – the fanotify technology is not used.
`StartupTraces`	Enables generation of trace files at application startup.	`true/yes` – Create trace files at application startup. `false/no` (default value) – Do not create trace files at application startup.
`RevealSensitiveInfoInTraces`	Display information in trace files that may contain personal data (for example, passwords).	`true/yes` (default value) — display information in application trace files that may contain personal data. `false/no` (default value) — do not display information that may contain personal data in trace files.
`AsyncTraces`	Enables asynchronous tracing, in which information is logged to trace files in asynchronously.	`true/yes` – enable asynchronous tracing. `false/no` (default value) – do not enable asynchronous tracing.

Setting	Description	Values
`CoreDumps`	Enables the creation of a dump file when application failure occurs.	`true/yes` to create a dump file when the application crashes. `false/no` (default value) not to create a dump file when the application crashes.
`CoreDumpsPath`	Path to the directory where the dump files are stored.	Default value: /var/opt/kaspersky/kess/common/dumps. Root privileges are required to access the default dump file directory.
`MinFreeDiskSpace`	The minimum amount of disk memory that will remain after writing a dump file, in megabytes.	Default value: 300.
`ScanMemoryLimit`	Limit on the application's use of memory in megabytes.	Default value: 8192. We recommend specifying a `ScanMemoryLimit` lower than the `MaxMemory` setting to avoid using up all available memory for scan tasks and causing an application restart.
`MachineId`	The user's unique device ID.	The value of the setting is filled in automatically during installation of the application.
`SocketPath`	The path to the socket for remote connection, through which, for example, the graphical interface and the kess-control utility are connected.	Default value: /var/run/bl4control.
`MaxInotifyWatches`	Limit on the number of subscriptions to changes in files and directories (user watches) in /proc/sys/fs/inotify/max_user_watches. If the setting in the operating system is less than the `MaxInotifyWatches` setting, the application changes the operating system setting to the `MaxInotifyWatches` value.	Default value: 299987.
`MaxInotifyInstances`	Limit on the number of subscriptions to changes in files and directories for one user set in /proc/sys/fs/inotify/max_user_instances. If the setting in the operating system is less than the `MaxInotifyInstances` setting, the application changes the operating system setting to the `MaxInotifyInstances` value.	Default value: 2047.
`ExecEnvMax`	The number of environment variables that the application captures from the command call.	Default value: 50.
`ExecArgMax`	Number of arguments that the application captures from the exec call.	Default value: 20.
`AdditionalDNSLookup`	Indicates use of a public DNS. If there are errors accessing servers through the system DNS, the application uses a public DNS. This is needed for updating application databases and maintaining device security. The application will use the following public DNSes in this order: 1. Google Public DNS™ (8.8.8.8). 2. Cloudflare® DNS (1.1.1.1). 3. Alibaba Cloud® DNS (223.6.6.6). 4. Quad9® DNS (9.9.9.9). 5. CleanBrowsing (185.228.168.168).	`true/yes` – Use a public DNS to access Kaspersky servers. `false/no` (default value) – Do not use a public DNS to access Kaspersky servers. The application's requests may contain domain addresses and the user's external IP address, since the application establishes a TCP/UDP connection with the DNS server. This information is necessary, for example, to check the certificate of a web resource when interacting via HTTPS. If the application is using a public DNS server, data processing rules are governed by the Privacy Policy of the corresponding service.
The **[Network]** section contains the following settings:		

Setting	Description	Values
`WtpFwMark`	A mark in the iptables rules for forwarding traffic to the application for processing by Web Threat Protection component. You may need to change this mark if a device with the application runs other software that uses the ninth bit of the TCP packet mask, and a conflict occurs.	A decimal value or hexadecimal number with the prefix 0x. Default value: 0x100. To avoid cyclical traffic scanning when the application is running on nodes used as part of distributed systems, such as Kubernetes, you need to change the values of the network packet marks using the higher bits. For example: 0x1000 or 0x10000 (the mark is limited to a 32-bit integer).
`NtpFwMark`	A mark in the iptables rules for forwarding traffic to the application for processing by Network Threat Protection component. You may need to change this mark if a device with the application runs other software that uses the ninth bit of the TCP packet mask, and a conflict occurs.	A decimal value or hexadecimal number with the prefix 0x. Default value: 0x200. To avoid cyclical traffic scanning when the application is running on nodes used as part of distributed systems, such as Kubernetes, you need to change the values of the network packet marks using the higher bits. For example: 0x2000 or 0x20000 (the mark is limited to a 32-bit integer).
`BypassFwMark`	A mark used to indicate packets created or scanned by the application, so that the application does not scan them again.	A decimal value or hexadecimal number with the prefix 0x. Default value: 0x400. To avoid cyclical traffic scanning when the application is running on nodes used as part of distributed systems, such as Kubernetes, you need to change the values of the network packet marks using the higher bits. For example: 0x4000 or 0x40000 (the mark is limited to a 32-bit integer).
`BypassNFlogMark`	A mark used to indicate packages created or scanned by the application to prevent them from being logged by the iptable utility.	A decimal value or hexadecimal number with the prefix 0x. Default value: 0x800. To avoid cyclical traffic scanning when the application is running on nodes used as part of distributed systems, such as Kubernetes, you need to change the values of the network packet marks using the higher bits. For example: 0x8000 or 0x80000 (the mark is limited to a 32-bit integer).
`ProxyRouteTable`	Number of the routing table.	Default value: 101.
The **[ScannerImpactStats]** section contains the following settings:		
`CollectFileScanStatistics`	Enables the tallying of statistics of file and process scanning by the File Threat Protection and Behavior Detection components.	`true/yes` (default value) enables the tallying of file and process scanning statistics. `false/no` disables the tallying of file and process scanning statistics.
`StatCollectionPeriod`	The time interval for which the application keeps a tally of file and process scanning statistics by the File Threat Protection and Behavior Detection components before saving the statistics to a trace file and reports.	Default value: 10 minutes.

Setting	Description	Values
`StatCollectionCount`	The number of files and processes counted by the application during the time interval specified by the `StatCollectionPeriod` parameter and that will be included in the report and trace files.	Default value: 10.
`ReportStatAmount`	The number of records to be written to reports on the most frequently scanned files and processes for the day.	Default value: 20.
The **[Watchdog]** section contains the following settings:		
`TimeoutAfterHeadshot`	Maximum time to wait for the kess process to finish from the moment the Watchdog server sends the HEADSHOT signal to the kess process.	Default value: 2 minutes.
`StartupTimeout`	The maximum time to wait for the application to start (in minutes), after which the Watchdog server starts the procedure for restarting the kess process.	Default value: 3 minutes.
`TimeoutAfterKill`	Maximum time to wait for the controlled kess process to complete from the moment the Watchdog server sends the SIGKILL signal to the kess process. If the kess process does not finish before this time elapses, the action specified by the --failed-kill setting is performed.	Default value: 2 days.
`PingInterval`	The interval in milliseconds with which the application attempts to send a PONG message to the Watchdog server in response to a received PING message.	Default value: 2000.
`MaxRestartCount`	Maximum number of consecutive unsuccessful attempts to start the application.	Default value: 5.
`ActivityTimeout`	Maximum time interval during which the application should send a message to the Watchdog server. If a message is not received from the application within this time interval, the Watchdog server begins the procedure to restart the kess process.	Default value: 2 minutes.
`ConnectTimeout`	Maximum time from the start of the kess process to the moment when a connection with the Watchdog server is established by the application. If the application does not establish a connection in this time interval, the Watchdog server begins the procedure to restart the kess process.	Default value: 3 minutes.
`RegisterTimeout`	Maximum time from the moment the application connects to the Watchdog server to the moment the server receives a REGISTER message.	Default value: 500 milliseconds.
`TimeoutAfterShutdown`	Maximum time to wait for the kess process to finish from the moment the Watchdog server sends the SHUTDOWN signal to the kess process.	Default value: 2 minutes.

Setting	Description	Values
MaxMemory	Limit on the use of resident memory by the kess process. If the managed process uses more resident memory than this limit, the Watchdog server begins the procedure to restart the kess process.	off – the resident set size is not limited. < value >% – a value between 1 and 100 percent of RAM. < value >MB – a value in megabytes. lowest/< value >%/< value >MB – the smaller of the percentage value and the absolute value in megabytes. highest/< value >%/< value >MB – the larger of the percentage value and the absolute value in megabytes. auto – up to 50% of available memory, but not less than 2GB and not more than 16GB. Default value: auto.
MaxVirtualMemory	Limit on the use of virtual memory by the kess process. If the managed process uses more virtual memory than this limit, the Watchdog server begins the procedure to restart the kess process.	off (default value) – The virtual memory size is not limited. < value >MB – a value in megabytes.
MaxSwapMemory	Limit on the size of the swap file of the kess process. If the swap file of the managed process exceeds this limit, the Watchdog server begins the procedure to restart the kess process.	off (default value) – The size of the swap file is not limited. < value >% – a value between 0 and 100 percent of RAM. < value >MB – a value in megabytes. lowest/< value >%/< value >MB – the smaller of the percentage value and the absolute value in megabytes. highest/< value >%/< value >MB – the larger of the percentage value and the absolute value in megabytes.
TrackProductCrashes	Enabling application stability monitoring. If application stability monitoring is enabled, the Watchdog server tracks the number of abnormal halts of the application.	true/yes – enable application stability monitoring. false/no (default value) – disable application stability monitoring.
ProductHealthLogFile	The path to the file used for application stability monitoring.	Default value: /var/opt/kaspersky/kess/private/kess_health.log.
WarnThreshold	Time interval (in seconds) in which the application must experience the specified number of abnormal halts before displaying a notification about unstable operation.	Default value: 3600.
WarnAfter_#_crash	Number of abnormal halts of the application that are required before displaying a notification about unstable application operation.	Default value: 10. If the value is 0, an unstable application notification is not displayed.
WarnRemovingThreshold	Time interval (in seconds) after which the application's unstable status will be cleared.	Default value: 86400.
SyscallHangProbePeriod	The period with which the Watchdog server calls the open and execve system functions and increments the success counters for these functions.	Default value: 3 seconds.
SyscallHangCheckPeriod	The period with which the Watchdog server checks the success counters for the open and execve functions. If the value of the counters is unchanged after this time, the Watchdog server starts the procedure for restarting the kess process.	Default value: 32 seconds.
DumpSuspendPeriod	The maximum time to wait in minutes for the creation of an application dump file, during which the Watchdog server suspends the checking of application activity. If the dump creation has not completed after this time, the Watchdog server starts the procedure for restarting the kess process.	Possible values: 1–30 minutes. Default value: 2.

Setting	Description	Values
`TimeoutAfterCoredump`	The maximum duration for which the Watchdog server suspends the sending of other signals to the kess process after sending the SIGTRAP signal as part of the abnormal process termination procedure.	Default value: 2 minutes. We recommend setting a `TimeoutAfterCoredump` greater than or equal to the `DumpSuspendPeriod`. You can use the following units to specify this setting: • `<n> day|days|d` for days • `<n> hour|hours|hr|h` for hours • `<n> minute|minutes|min|m` for minutes • `<n> second|seconds|sec|s` for seconds • `<n> [msec|ms]` for milliseconds If no unit is specified, the application interprets the value as milliseconds.

If a parameter is absent in the file or an incorrect parameter value is specified, the application will use the default value.

Default settings for command line tasks

This section contains the default options for all predefined tasks that can be used to manage Kaspersky Embedded Systems Security on the command line.

The *Rollback* and *License* tasks have no settings.

Default settings for the File_Threat_Protection task (ID:1)

```
ScanArchived=No

ScanSfxArchived=No

ScanMailBases=No

ScanPlainMail=No

SkipPlainTextFiles=No

SkipDatabaseJournals=No

TimeLimit=60

SizeLimit=0

ActionOnThreat=DisinfectDeleteIfNotPossible

UseExcludeMasks=No
```

```
UseExcludeThreats=No

ReportCleanObjects=No

ReportPackedObjects=No

ReportUnprocessedObjects=No

UseAnalyzer=Yes

HeuristicLevel=Recommended

UseIChecker=Yes

ScanByAccessType=SmartCheck

[ScanScope.item_0000]

AreaDesc=All objects

UseScanArea=Yes

Path=/

AreaMask.item_0000=*
```

Default settings for the Scan_My_Computer task (ID:2)

```
ScanFiles=Yes

ScanBootSectors=Yes

ScanComputerMemory=Yes

ScanStartupObjects=Yes

ScanArchived=Yes

ScanSfxArchived=Yes

ScanMailBases=No

ScanPlainMail=No

TimeLimit=0

SizeLimit=0

ActionOnThreat=DisinfectDeleteIfNotPossible

UseExcludeMasks=No

UseExcludeThreats=No
```

```
ReportCleanObjects=No

ReportPackedObjects=No

ReportUnprocessedObjects=No

UseAnalyzer=Yes

HeuristicLevel=Recommended

UseIChecker=Yes

UseGlobalExclusions=Yes

UseOASExclusions=Yes

DeviceNameMasks.item_0000=/**

[ScanScope.item_0000]

AreaDesc=All objects

UseScanArea=Yes

Path=/

AreaMask.item_0000=*
```

Default settings for the Scan_File task (ID:3)

```
ScanFiles=Yes

ScanBootSectors=No

ScanComputerMemory=No

ScanStartupObjects=No

ScanArchived=Yes

ScanSfxArchived=Yes

ScanMailBases=No

ScanPlainMail=No

TimeLimit=0

SizeLimit=0

ActionOnThreat=DisinfectDeleteIfNotPossible

UseExcludeMasks=No
```

```
UseExcludeThreats=No

ReportCleanObjects=No

ReportPackedObjects=No

ReportUnprocessedObjects=No

UseAnalyzer=Yes

HeuristicLevel=Recommended

UseIChecker=Yes

UseGlobalExclusions=Yes

UseOASExclusions=Yes

DeviceNameMasks.item_0000=/**

[ScanScope.item_0000]

AreaDesc=All objects

UseScanArea=Yes

Path=/

AreaMask.item_0000=*
```

Default settings for the Critical_Areas_Scan task (ID:4)

```
ScanFiles=No

ScanBootSectors=Yes

ScanComputerMemory=Yes

ScanStartupObjects=Yes

ScanArchived=Yes

ScanSfxArchived=Yes

ScanMailBases=No

ScanPlainMail=No

TimeLimit=0

SizeLimit=0

ActionOnThreat=DisinfectDeleteIfNotPossible
```

```
UseExcludeMasks=No

UseExcludeThreats=No

ReportCleanObjects=No

ReportPackedObjects=No

ReportUnprocessedObjects=No

UseAnalyzer=Yes

HeuristicLevel=Recommended

UseIChecker=Yes

UseGlobalExclusions=Yes

UseOASExclusions=Yes

DeviceNameMasks.item_0000=/**

[ScanScope.item_0000]

AreaDesc=All objects

UseScanArea=Yes

Path=/

AreaMask.item_0000=*
```

Default settings for the Update task (ID:6)

```
ConnectionTimeout=10

ApplicationUpdateMode=DownloadOnly

[Sources.item_0000]

Enabled=Yes

SourceType=KLServers

URL=
```

Default settings for the System_Integrity_Monitoring task (ID:11)

```
UseExcludeMasks=No

[ScanScope.item_0000]
```

```
AreaDesc=Kaspersky internal objects

UseScanArea=Yes

Path=/opt/kaspersky/kess/

AreaMask.item_0000=*
```

Default settings for the Firewall_Management task (ID:12)

```
DefaultIncomingAction=Allow

DefaultIncomingPacketAction=Allow

OpenNagentPorts=Yes

[NetworkZonesTrusted]

[NetworkZonesLocal]

[NetworkZonesPublic]
```

Default settings for the Anti_Cryptor task (ID:13)

```
ActionOnDetect=Block

BlockTime=30

UseExcludeMasks=No

[ScanScope.item_0000]

AreaDesc=All shared directories

UseScanArea=Yes

Path=AllShared

AreaMask.item_0000=*
```

Default settings for the Web_Threat_Protection task (ID:14)

```
UseTrustedAddresses=Yes

ActionOnDetect=Block

CheckMalicious=Yes

CheckPhishing=Yes
```

```
UseHeuristicForPhishing=Yes

CheckAdware=No

CheckOther=No
```

Default settings for the Device_Control task (ID:15)

```
OperationMode=Block

AllowToRequestTemporaryAccess=No

[DeviceClass]

HardDrive=DependsOnBus

RemovableDrive=DependsOnBus

Printer=DependsOnBus

FloppyDrive=DependsOnBus

OpticalDrive=DependsOnBus

Modem=DependsOnBus

TapeDrive=DependsOnBus

MultifuncDevice=DependsOnBus

SmartCardReader=DependsOnBus

PortableDevice=DependsOnBus

WiFiAdapter=DependsOnBus

NetworkAdapter=DependsOnBus

BluetoothDevice=DependsOnBus

ImagingDevice=DependsOnBus

SerialPortDevice=DependsOnBus

ParallelPortDevice=DependsOnBus

InputDevice=DependsOnBus

SoundAdapter=DependsOnBus

[DeviceBus]

USB=Allow
```

```
FireWire=Allow

[Schedules.item_0000]

ScheduleName=Default

DaysHours=All

[HardDrivePrincipals.item_0000]

Principal=\Everyone

[HardDrivePrincipals.item_0000.AccessRules.item_0000]

UseRule=Yes

ScheduleName=Default

Access=Allow

[RemovableDrivePrincipals.item_0000]

Principal=\Everyone

[RemovableDrivePrincipals.item_0000.AccessRules.item_0000]

UseRule=Yes

ScheduleName=Default

Access=Allow

[FloppyDrivePrincipals.item_0000]

Principal=\Everyone

[FloppyDrivePrincipals.item_0000.AccessRules.item_0000]

UseRule=Yes

ScheduleName=Default

Access=Allow

[OpticalDrivePrincipals.item_0000]

Principal=\Everyone

[OpticalDrivePrincipals.item_0000.AccessRules.item_0000]

UseRule=Yes

ScheduleName=Default
```

```
Access=Allow
```

Default settings for the Removable_Drives_Scan task (ID:16)

```
ScanRemovableDrives=NoScan

ScanOpticalDrives=NoScan

BlockDuringScan=No
```

Default settings for the Network_Threat_Protection task (ID:17)

```
ActionOnDetect=Block

BlockAttackingHosts=Yes

BlockDurationMinutes=60

UseExcludeIPs=No
```

Default settings for the Behavior_Detection task (ID:20)

```
UseTrustedPrograms=No

TaskMode=Block

UseTelemetryMonopolyMode=Yes

TelemetrySource=Default
```

Default settings for the Application_Control task (ID:21)

```
AppControlMode=DenyList

AppControlRulesAction=ApplyRules

UseTrustedCustomCerts=Yes
```

Default settings for the Inventory_Scan task (ID:22)

```
ScanScripts=Yes

ScanBinaries=Yes

ScanAllExecutable=Yes

GoldenImageAction=DoNothing
```

```
[ScanScope.item_0000]

AreaDesc=All objects

UseScanArea=Yes

Path=/usr/bin

AreaMask.item_0000=*
```

Default settings of the BadUSB Attack Prevention task (ID: 37)

```
MaxAuthAttempts=5

TimeoutAfterAttempts=5

ProhibitOnScreenKeyboard=No
```

General application settings

General application settings define the operation of the application as a whole and the operation of individual functions.

General application settings

Setting	Description	Values
`SambaConfigPath`	Directory that stores the Samba configuration file. The Samba configuration file is required to ensure that the `AllShared` or `Shared:SMB` values can be used for the `Path` setting. The application must be restarted after this setting is changed.	The standard directory of the SAMBA configuration file on the computer is specified by default. Default value: /etc/samba/smb.conf.
`NfsExportPath`	The directory where the NFS configuration file is stored. The NFS configuration file is required to ensure that the `AllShared` or `Shared:NFS` values can be used for the `Path` setting. The application must be restarted after this setting is changed.	The standard directory of the NFS configuration file on the computer is specified by default. Default value: /etc/exports.
`TraceLevel`	Enable application tracing and the level of detail in the trace files.	`Detailed` – Generate a detailed trace file. `MediumDetailed` – Generate a trace file that contains informational messages and error messages. `NotDetailed` – Generate a trace file that contains error messages. `None` (default value) — Do not generate a trace file.

Setting	Description	Values
`TraceFolder`	The directory that stores the application trace files. The application must be restarted after this setting is changed.	Default value: /var/log/kaspersky/kess. Root privileges are required to access the default trace files directory. If you specify a different directory, make sure that only the root has read and write permissions to this directory (`700` or `rwx--------` permissions).
`TraceMaxFileCount`	Maximum number of application trace files. The application must be restarted after this setting is changed.	1–10000 Default value: 10.
`TraceMaxFileSize`	Specifies the maximum size of an application trace file (in megabytes). The application must be restarted after this setting is changed.	1–1000 Default value: 500.
`BlockFilesGreaterMaxFileNamePath`	Blocks access to files for which the full path length exceeds the defined settings value specified in bytes. If the length of the full path to the scanned file exceeds the value of this setting, scan tasks skip this file during scanning. This setting only applies to operating systems that do not support fanotify. After changing the value of this setting, the File Threat Protection task needs to be restarted.	4096–33554432 Default value: 16384.
`DetectOtherObjects`	Enable detection of legitimate applications that intruders can use to compromise devices or data.	`Yes`: enable detection of legitimate applications that intruders can use to compromise devices or data. `No` (default): disable detection of legitimate applications that intruders can use to compromise devices or data.
`NamespaceMonitoring`	Enabling the use of the namespace mechanism, which also allows scanning files in containers and mandatory access control sessions of the Astra Linux operating system. The application does not scan namespaces or containers unless components for managing namespaces are installed in the operating system.	`Yes` (default value) – Enable the namespace mechanism. `No` – disable the namespace mechanism.

Setting	Description	Values
`FileBlockDuringScan`	Enabling the file operation intercept mode with blocking access to files for the duration of the scan. The file operation interception mode affects the File Threat Protection and Device Control components.	`Yes` (default value) to block access to files for the duration of the scan. `No` to allow access to files during the scan. Requests to any file is allowed, scanning is done asynchronously. This file operation interception mode has less impact on the system performance, but there is a risk that a threat in a file will not be disinfected or deleted if the file can, for example, change its name during a scan before the application makes a decision on the status of the file. However, the File Threat Protection and Device Control components operate in notify mode, regardless of which mode is configured.
`UseKSN`	Configure Kaspersky Security Network for the application.	`Basic` — Enable use of Kaspersky Security Network in standard mode. `Extended` — Enable use of Kaspersky Security Network in extended mode. `No` (default value) — Do not use the Kaspersky Security Network.
`CloudMode`	Enable cloud mode. Cloud mode is available if use of KSN is enabled. If you plan to use cloud mode, make sure KSN is available on your device.	`Yes` — enable the mode in which Kaspersky Embedded Systems Security uses a lightweight version of the malware databases. `No` (default value) – use the full version of the malware databases. Cloud mode is disabled automatically if use of KSN is disabled.
`UseMDR`	Enabling the Managed Detection and Response component for integration with Kaspersky Managed Detection and Response. If you enable the MDR component, and Kaspersky Security Network is disabled or used in Basic mode, an error message is displayed and the MDR component is not enabled. To enable the MDR component, you must enable KSN in Extended mode.	`Yes` to enable the Managed Detection and Response component. `No` (default value) – disable the Managed Detection and Response component.
`UseProxy`	Enables the use of a proxy server by Kaspersky Embedded Systems Security components.	`Yes` — Enable the use of a proxy server. `No` (default) — Disable the use of a proxy server.
`ProxyServer`	Address of the proxy server used to connect to the internet. When connecting via an HTTP proxy, we recommend to use a separate account that is not used to sign in to other systems. An HTTP proxy uses an insecure connection, and the account may be compromised.	Address of the proxy server in one of the following formats: • `[<connection protocol>://]<IP address of the proxy server>:<port>`, if no authentication is required to connect to the proxy server; • `[<connection protocol>://][<user name>:[<password>]@]<IP address of the proxy server>:<port>`, if authentication is required when connecting to the proxy server. Connecting to a proxy server over HTTPS is not supported.
`ProxyBypass`	List of addresses for which the proxy server is to be bypassed.	The address is specified in the `< address >[:< port number >` format. To specify addresses, you can use masks (`*` symbols) and comments (after a \ symbol).

Setting	Description	Values
`MaxEventsNumber`	The maximum number of events stored by the application. When the specified number of events is exceeded, the application deletes the oldest events.	Default value: 500000. If the value is 0, new events are not saved, and old events are deleted.
`LimitNumberOfScanFileTasks`	The maximum number of custom scan tasks that a non-privileged user can simultaneously start on the device. This setting does not limit the number of tasks that a user with root privileges can start.	0–100000 0 means a non-privileged user cannot start custom scan tasks. Default value: 5.
`UseSyslog`	Enables logging of application events to syslog. Root privileges are required to access syslog.	`Yes`: enable saving events to syslog. `No` (default): disable saving events to syslog.
`EventsStoragePath`	Path to the main file of the event log database. Root privileges are required to access the default event database.	Default value: /var/opt/kaspersky/kess/private/storage/events.db.
`ExcludedMountPoint.item_#`	The mount point to exclude from the scan scope. The exclusion applies to the operation of the File Threat Protection and Anti-Cryptor components, the Removable Drives Scan task, and is also configured for scan tasks of the ODS type. You can specify several mount points to be excluded from scans. Mount points must be specified in the same way as they are displayed in the `mount` command output. The `ExcludedMountPoint.item_#` setting is left unspecified by default.	`AllRemoteMounted` — Exclude all remote directories mounted on the device using SMB and NFS protocols from file operation interception. `Mounted:NFS` — Exclude all remote directories mounted on the device using the NFS protocol from file operation interception. `Mounted:SMB` — Exclude all remote directories mounted on the device using the SMB protocol from file operation interception. `Mounted:<file system type>` — Exclude all mounted directories with the specified file system type from file operation interception. `/mnt` — Exclude objects in the /mnt mount point (including subdirectories) from file operation interception. This directory is used as the temporary mount point for removable drives. `<path that contains the /mnt/user* or /mnt/**/user_share>` mask — Exclude objects in mount points whose names contain the specified **mask** from file operation interception. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. To exclude the mount point `/dir`, you need to specifically indicate `/dir` (no asterisk). The mask `/dir/*` excludes all mount points at the level below `/dir` but not `/dir` itself. The `/dir/**` mask excludes all mount points below the level of `/dir` but not `/dir` itself. You can use a single `?` character to represent any one character in the file or directory name.

Setting	Description	Values
`MemScanExcludedProgramPath.item_#`	Exclude process memory from scans. The application does not scan the memory of the indicated process.	`< full path to process >` – Do not scan the process in the specified local directory. You can use **masks** to specify the path. You can use the `*` (asterisk) character to create a file or directory name mask. You can indicate a single `*` character to represent any set of characters (including an empty set) preceding the `/` character in the file or directory name. For example, `/dir/*/file` or `/dir/*/*/file`. You can indicate two consecutive `*` characters to represent any set of characters (including an empty set and the `/` character) in the file or directory name. For example, `/dir/**/file*/` or `/dir/file**/`. The `**` mask can be used only once in a directory name. For example, `/dir/**/**/file` is an incorrect mask. You can use a single `?` character to represent any one character in the file or directory name.
`UseOnDemandCPULimit`	Enables CPU usage limits for task types: *ODS* and *InventoryScan*.	`Yes`: enable the CPU usage limit for *ODS* and *InventoryScan* tasks. `No` (default): disable CPU usage limits for tasks.
`OnDemandCPULimit`	Maximum utilization of all CPU cores (as a percentage) for task types: *ODS* and *InventoryScan*.	Integer value from 10 to 100. Default value: 100.
`BackupDaysToLive`	Time period for storing objects in Backup (in days). After the specified time has elapsed, the application deletes the oldest backup copies of files. To remove the object retention limit, set 0.	0 to 10000 0–unlimited retention. Default value: 30.
`BackupSizeLimit`	Maximum Backup size (in MB). When the maximum Backup storage size is reached, the application deletes the oldest backup copies of files. To remove the Backup size limit, set 0.	0 to 4000 0–unlimited size. Default value: 0.
`QuarantineBackupFolder`	The path to the directory where backup copies of files and quarantined files are stored. You can change the default directory. You can use directories on any device as the Backup storage. It is not recommended to assign directories that are located on remote devices, such as those mounted via the Samba and NFS protocols. If the specified directory does not exist or is unavailable, the application uses the default directory.	Default value: /var/opt/kaspersky/kess/common/objects-backup/ Root privileges are required for access to the default directory.
`QuarantineFillingNotification`	The percentage of Quarantine that must be full to generate an event about Quarantine being full. To disable generating events about Quarantine being full, enter 0.	0 to 100 0 – generating events about Quarantine being full is disabled. Default value: 90.
`QuarantineSizeLimit`	Maximum size of the Quarantine (in megabytes). When the maximum Quarantine size is reached, the application deletes the oldest files. To make the size of Quarantine unlimited, enter 0.	0 to 4000 0 means the Quarantine size is not limited. Default value: 200.

Setting	Description	Values
ShowPopUpNotifications	Enables the display of pop-up notifications in the graphical user interface.	Yes (default value) – show pop-up notifications in the graphical user interface. No – do not show pop-up notifications in the graphical user interface.
PowerSaveMode	Canceling scheduled tasks on a device running on battery power.	Yes (default value) skips scheduled tasks when the device is running on battery power. No runs scheduled tasks regardless of whether the device is running on battery power or not.

Network traffic scan settings

Network traffic scan settings

Setting	Description	Values
EncryptedConnectionsScan	Enables or disables the scanning of encrypted connections. For the FTP protocol, encrypted connections scan is disabled by default.	Yes (default value) — Enable encrypted connection scans. No — Disable encrypted connection scans. The application does not decrypt the encrypted traffic.
EncryptedConnectionsScanErrorAction	Specifies the action to perform when an encrypted connection scan error occurs on a website.	AddToAutoExclusions (default value) — Add the domain where an error occurred to the list of domains with scan errors. The application will not monitor encrypted network traffic when this domain is visited. Disconnect — Block the network connection.
CertificateVerificationPolicy	Specifies the way Kaspersky Embedded Systems Security checks certificates. If a certificate is self-signed, the application does not perform the additional verification.	FullCheck (default value) — The application uses the Internet to check and download the missing chains that are required to verify a certificate. LocalCheck — The application does not use the Internet to verify a certificate.
UntrustedCertificateAction	The action to take when an unconfirmed certificate is detected.	Allow (default value) — Allow network connections established while visiting a domain with an untrusted certificate. Block — Block network connections established while visiting a domain with an untrusted certificate.
ManageExclusions	Using exclusions when scanning encrypted traffic.	Yes — Do not scan websites specified under [Exclusions.item_#] (see below). No (default value) — Scan all websites.
MonitorNetworkPorts	Specifies the way Kaspersky Embedded Systems Security monitors network ports.	Selected (default value) — Monitor only network ports specified in the [NetworkPorts.item_#] section (see below). All — Monitor all network ports. Specifying this value may significantly increase an operating system load.
NetInterceptorMode	Traffic interception mode by the Kaspersky Embedded Systems Security application.	UseEBPF — use the eBPF technology to intercept traffic. UseIPTables (default value) — use the iptables utility to intercept traffic. To change the traffic interception mode, you need to restart the application. Restarting breaks established network connections.

Setting	Description	Values
The **[Exclusions.item_#]** section contains domains excluded from scans. The application does not scan connections established when visiting specified domains.		
`DomainName`	Specifies the domain name.	You can use masks when specifying a domain.
The **[IncomingEndpointsExclusions.item_#]** section contains exclusions for incoming traffic.		
`RemoteIp`	IP address of the remote device, traffic from which is excluded from interception.	
`DestinationPort`	Port on the protected device, traffic to which from remote devices is excluded from interception.	
The **[OutgoingEndpointsExclusions.item_#]** section contains exclusions for outgoing traffic.		
`RemoteIp`	IP address of the remote device, traffic to which is excluded from interception.	
`DestinationPort`	Port on the remote device, traffic to which from the protected device is excluded from interception.	
The **[NetworkPorts.item_#]** section contains the network ports monitored by the application.		
`PortName`	Network port description.	
`Port`	Network port numbers to be monitored by the application.	`1 – 65535`. The default value is not defined.

Tasks schedule settings

The table below describes the task schedule settings that you can manage for all application tasks.

Task schedule settings

Setting	Description	Values
`RuleType`	Task launch schedule.	`Once`: run the task once. `Monthly`: run the task on the specified day and time every month. `Weekly`: run the task on the specified day and time every week. `Daily`: run the task regularly, at the specified interval in days. `Hourly`: run the task regularly, at the specified interval in hours, starting on the specified date and time. `Minutely`: run the task regularly, at the specified interval in minutes, starting at the specified time. `Manual` – start the task manually. `PS` – start the task after starting the application. `BR` – start the task after the application databases have been updated.
`StartTime`	Task start date and time. The `StartTime` option is required if the `RuleType` option is set to one of the following: Once, Monthly, Weekly, Daily, Hourly, or Minutely.	`[< year >/< month >/< day of the month >] [ hh ]:[ mm ]:[ ss ]; [< day of the month >| < day of the week >]; [< start periodicity >]`
`RandomInterval`	A time interval from 0 to the specified value (in minutes), which will be added to the task start time to avoid starting tasks at the same time.	Default value: 99 minutes.

Setting	Description	Values
`RunMissedStartRules`	Runs a missed task after the application is started.	`Yes` (default value) – enable running a missed task after the application is launched. `No`: does not enable running a missed task after the application starts.
`UseWorkingTimeout`	Stops the task upon reaching the maximum task execution time specified by the `WorkingTimeout` setting. The task will be stopped even if it is not completed. The next launch of the task will be performed according to the schedule.	`Yes` – Stop the task when the maximum task execution time has been reached. `No` (default value) – Do not stop the task when the maximum task execution time has been reached.
`WorkingTimeout`	The maximum task execution time (in minutes) after which the application stops executing the task if `UseWorkingTimeout=yes`.	Default value: 120 minutes.

Appendix 3. Command line return codes

Kaspersky Embedded Systems Security has the following command line return codes:

0 means command/task completed successfully

1 means general error in command syntax

2 – error in passed application settings.

64 – Kaspersky Embedded Systems Security is not running.

66 – application databases are not loaded (used only by the `kess-control --app-info` command).

67 – activation 2.0 ended with an error due to network problems.

68 means the command cannot be executed because the application is managed by a policy.

69 – the application is located in the Amazon Paid Ami infrastructure.

70 – an attempt to start a running task, delete a running task, change the settings of a running task, stop a stopped task, pause a suspended task, or resume a running task.

71 – Kaspersky Security Network Statement has not been accepted.

72 – threats were detected by the Custom scan task.

73 – attempt to set Application Control task settings that affect the operation of the Kaspersky Security Network application without confirming them using the `--accept` option.

74 – Kaspersky Embedded Systems Security must be restarted after an update.

75 means the device must be restarted.

76 — connection prohibited, as only users with root rights should have write access to the specified path.

77 — the specified license key is already in use on the device.

80 – the update download operation failed due to network problems.

128 means unknown error.

65 – all other errors.

Appendix 4. Settings for interaction with external services

Kaspersky Embedded Systems Security uses the following network settings and encryption settings for interaction with external services.

Settings for interaction with external services

Network address of the service and protocol	Description
`activation-v2.kaspersky.com/activationservice/activationservice.svc` Protocol: `HTTPS` Port: `443`	Application activation.

`s00.upd.kaspersky.com`
`s01.upd.kaspersky.com`
`s02.upd.kaspersky.com`
`s03.upd.kaspersky.com`
`s04.upd.kaspersky.com`
`s05.upd.kaspersky.com`
`s06.upd.kaspersky.com`
`s07.upd.kaspersky.com`
`s08.upd.kaspersky.com`
`s09.upd.kaspersky.com`
`s10.upd.kaspersky.com`
`s11.upd.kaspersky.com`
`s12.upd.kaspersky.com`
`s13.upd.kaspersky.com`
`s14.upd.kaspersky.com`
`s15.upd.kaspersky.com`
`s16.upd.kaspersky.com`
`s17.upd.kaspersky.com`
`s18.upd.kaspersky.com`
`s19.upd.kaspersky.com`
Protocol: `HTTPS`
Port: `443`
	Updating application databases and modules.

Network address of the service and protocol	Description
p00.upd.kaspersky.com p01.upd.kaspersky.com p02.upd.kaspersky.com p03.upd.kaspersky.com p04.upd.kaspersky.com p05.upd.kaspersky.com p06.upd.kaspersky.com p07.upd.kaspersky.com p08.upd.kaspersky.com p09.upd.kaspersky.com p10.upd.kaspersky.com p11.upd.kaspersky.com p12.upd.kaspersky.com p13.upd.kaspersky.com p14.upd.kaspersky.com p15.upd.kaspersky.com p16.upd.kaspersky.com p17.upd.kaspersky.com p18.upd.kaspersky.com p19.upd.kaspersky.com downloads.kaspersky-labs.com Protocol: HTTP Port: 80	Updating application databases and modules.
downloads.upd.kaspersky.com Protocol: HTTPS Port: 443	• Updating application databases and modules. • Checking access to Kaspersky servers. If there are errors accessing servers through the system DNS, the application uses a public DNS. This is needed for updating application databases and maintaining device security. The application will use the following public DNSes in this order: 1. Google Public DNS (8.8.8.8). 2. Cloudflare DNS (1.1.1.1). 3. Alibaba Cloud DNS (223.6.6.6). 4. Quad9 DNS (9.9.9.9). 5. CleanBrowsing (185.228.168.168). The application's requests may contain domain addresses and the user's external IP address, since the application establishes a TCP/UDP connection with the DNS server. This information is necessary, for example, to check the certificate of a web resource when interacting via HTTPS. If the application is using a public DNS server, data processing rules are governed by the Privacy Policy of the corresponding service.
ds.kaspersky.com Protocol: HTTPS Port: 443	Using Kaspersky Security Network.

Network address of the service and protocol	Description
ksn-a-stat-geo.kaspersky-labs.com ksn-file-geo.kaspersky-labs.com ksn-verdict-geo.kaspersky-labs.com ksn-url-geo.kaspersky-labs.com ksn-a-p2p-geo.kaspersky-labs.com ksn-cinfo-geo.kaspersky-labs.com Protocol: Any Port: 443, 1443	Using Kaspersky Security Network.
click.kaspersky.com Protocol: HTTPS Port: 443	Following links from the user interface.
crl.kaspersky.com ocsp.kaspersky.com Protocol: HTTP Port: 80	Public Key Infrastructure (PKI). These servers (crl.kaspersky.com) are part of the public key infrastructure (PKI) and are required for verifying the validity status of Kaspersky digital signature certificates. The CRL server provides a list of revoked certificates. The OCSP service allows querying the status of a specific certificate in real time. These components maintain the security of interaction with digital certificates and provide protection against possible attacks.

Sources of information about Kaspersky Embedded Systems Security

Kaspersky Embedded Systems Security page on the Kaspersky website

On the Kaspersky Embedded Systems Security page, you can view general information about the application, its functions, and its features.

Discussion of Kaspersky applications in the Forum

If your question does not require an urgent answer, you can discuss it with Kaspersky experts and other users in our Forum.

In the Forum, you can view existing topics, post your own comments, and create new discussion topics.

Glossary

Activating the application

Switching the application to the fully functional mode. Application activation is performed during or after the installation of the application. You need an activation code or a key file to activate the application.

Active key

A key that is currently used by the application.

Active policy

Policy currently used by the application to control data leaks. The application can use several policies at the same time.

Administration group

A set of devices combined in Kaspersky Security Center in accordance with the functions they perform and the set of Kaspersky applications installed on them. Devices are grouped to simplify administration as the group of devices can be managed as a single entity. An administration group can include other groups. For each application installed in the administration group, group policies and group tasks can be created.

Administration Server

A component of Kaspersky Security Center that centrally stores information about all Kaspersky applications installed within the corporate network. It can also be used to manage these applications.

Application databases

Databases that contain information about computer security threats known to Kaspersky as of when the databases are released. Application databases are created by Kaspersky experts and updated hourly.

Application settings

Application settings that are common to all types of tasks and govern the overall operation of the application, such as application performance settings, reporting settings, and backup settings.

Database of malicious web addresses

A list of web addresses whose content may be considered to be dangerous. The list is created by Kaspersky specialists. It is regularly updated and is included in the Kaspersky application distribution kit.

Database of phishing web addresses

A list of web addresses which Kaspersky specialists have determined to be phishing-related. The database is regularly updated and is part of the Kaspersky application distribution kit.

Disinfection

A method of processing infected objects that results in complete or partial recovery of data. Not all infected objects can be disinfected.

Exclusion

Exclusion is an object excluded from the Kaspersky application scan. You can exclude from scan files of certain formats, file masks, a certain area (for example, a folder or an application), application processes, or objects by name, according to the Virus Encyclopedia classification. Each task can be assigned a set of exclusions.

False positive

A situation when a Kaspersky application considers a non-infected object to be infected because the object's code is similar to that of a virus.

File mask

Representation of a file name using wildcards. The standard wildcards used in file masks are * and ?, where * is any number of any characters, and ? is any single character.

Group policy

See Policy.

Group task

A task assigned to an administration group and performed on all managed devices included in this administration group.

Infected object

An object which includes a portion of code that completely matches the part of a known malware code. Kaspersky experts do not recommend accessing such objects.

Kaspersky update servers

Kaspersky HTTP and FTP servers from which Kaspersky applications download database and application module updates.

License

A time-limited right to use the application, granted under the End User License Agreement.

License certificate

A document that Kaspersky transfers to the user together with the key file or activation code. It contains information about the license granted to the user.

Policy

A policy determines the application settings and manages the access to configuration of an application installed on devices within an administration group. An individual policy must be created for each application. You can create an unlimited number of various policies for applications installed on the devices in each administration group, but only one policy can be applied to each application at a time within an administration group.

Proxy server

A computer network service which allows users to make indirect requests to other network services. First, a user connects to a proxy server and requests a resource (e.g., a file) located on another server. Then the proxy server either connects to the specified server and obtains the resource from it or returns the resource from its own cache (if the proxy has its own cache). In some cases, a user's request or a server's response can be modified by the proxy server for certain purposes.

Reserve key

A key that certifies the right to use the application but is not currently being used.

Startup objects

A set of applications needed for the operating system and software that is installed on the computer to start and operate correctly. These objects are executed every time the operating system is started. There are viruses capable of infecting such objects specifically, which may lead, for example, to blocking of operating system startup.

Subscription

Enables use of the application with the selected settings (expiration date and the number of devices). You can pause or resume your subscription, renew it automatically, or cancel it.

Trusted device

Device that can be fully accessed at any time by the users listed under the trusted device settings.

Information about third-party code

Information about third-party code is contained in the file legal_notices.txt located in the application installation folder.

Trademark notices

Registered trademarks and service marks are the property of their respective owners.

AMD64 is a trademark or registered trademark of Advanced Micro Devices, Inc.

Amazon is a trademark of Amazon.com, Inc. or its affiliates.

Mac and FireWire are trademarks of Apple Inc.

Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

The Bluetooth word, mark and logos are owned by Bluetooth SIG, Inc.

Ubuntu and LTS are registered trademarks of Canonical Ltd.

Cloudflare, the Cloudflare logo, and Cloudflare Workers are trademarks and/or registered trademarks of Cloudflare, Inc. in the United States and other jurisdictions.

The Grafana Word Mark and Grafana Logo are either registered trademarks/service marks or trademarks/service marks of Coding Instinct AB, in the United States and other countries and are used with Coding Instinct's permission. We are not affiliated with, endorsed or sponsored by Coding Instinct, or the Grafana community.

Docker and the Docker logo are trademarks or registered trademarks of Docker, Inc. in the United States and/or other countries. Docker, Inc. and other parties may also have trademark rights in other terms used herein.

Chrome, Google Public DNS are trademarks of Google LLC.

EulerOS is a trademark of Huawei Technologies Co., Ltd.

Core is a trademark of Intel Corporation or its subsidiaries.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft, Active Directory, Outlook, Visual C++, and Windows are trademarks of the Microsoft group of companies.

Oracle, Java, and JavaScript are registered trademarks of Oracle and/or its affiliates.

Python is a trademark or registered trademark of the Python Software Foundation.

Red Hat, Red Hat Enterprise Linux, and CentOS are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Debian is a registered trademark of Software in the Public Interest, Inc.

SUSE is a registered trademark of SUSE LLC in the United States and other countries.

Kubernetes is a registered trademark of The Linux Foundation in the United States and other countries.

Rocky Linux is a trademark of The Rocky Enterprise Software Foundation.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

Zabbix is a registered trademark of Zabbix SIA.