kaspersky

# Distributed ASAP On-Premise deployment

Application version: 1.1

kaspersky

Dear user,

Thank you for entrusting your security needs to us. We hope that this document will help you in your work and will provide answers to most of your questions.



Document revision date: November 5, 2024



https://www.kaspersky.com
https://help.kaspersky.com
https://support.kaspersky.com

Learn about Kaspersky at https://www.kaspersky.com/about/company

kaspersky

# Content

kaspersky

# What is ASAP?

The Kaspersky Automated Security Awareness Platform (ASAP) is an online training platform helping users learn information security rules and related threats in their everyday lives, and using practical examples to teach how to combat them.

Training helps develop all the necessary knowledge and skills in detail. The full training course includes the assimilation and consolidation of more than 350 elementary skills.

Training is broken down into units. Every training unit focuses on a specific topic at the program's corresponding level of difficulty. Training units contain several lessons each with an average duration of 5-10 minutes, which are then reinforced via repetition, tests and simulated phishing attacks during training on the topics (where applicable).

The platform can be deployed as standard or fault-tolerant:

- Standard: all data is stored on the same server.
- Fault-tolerant: data is stored in a cluster on multiple servers duplicating each other. One server is also dedicated for cluster management and data synchronization between servers.

## Distribution kit

The distribution kit includes the following files:

- Archive for installing ASAP On-Premise components
- Files with version information (release notes)

The files are available for download on the Kaspersky website:

- Distribution package – an archive with a set of files for deploying ASAP On-Premise components in your organization's infrastructure.
- Distribution package signature (see the "Downloading an XML file with the distribution package signature" section on p. 4) – digital signature of the ASAP On-Premise platform distribution package.
  You can use it to make sure that you have downloaded the correct file.
- Distribution package signature checker (see the "Checking the integrity of the distribution package" section on p. 5) – script for checking the integrity of the platform distribution package you downloaded.

## Downloading an XML file with the distribution package signature

► *To download an XML file with the distribution package signature:*

1. On the Kaspersky website, find the **Distribution package signature** item and click the nearby **Download** button.

   An XML file with the distribution package signature opens, but the browser displays an error message rather than the file content.

2. Save the opened XML file to your local drive. For example, in Google Chrome™, you need to press the **CTRL+S** key combination and select the directory where you want to place the file.

You have downloaded an XML file with the distribution package signature. It can be used to verify the authenticity of the downloaded KASAP On-Premise distribution package.

## Checking the integrity of the distribution package

► *To check the integrity of the downloaded platform distribution package:*

1. Download the platform distribution archive, XML file with the distribution package signature, and script for checking the integrity of the distribution package into one directory.
2. Change the permissions on the integrity_checker script to grant execution permissions to all users.

   For example, using the command `chmod 755 integrity_checker`.

3. Run the integrity_checker script and wait for it to finish executing.

If the integrity of the distribution package is successfully verified, the screen will display `SUCCEEDED`.

kaspersky

# Hardware and software requirements

- Minimum amount of RAM: 16 GB.
- Processor: 8 cores 16 threads with support for AVX and SSE 4.2.

  You can check for AVX and SSE 4.2 support using the command `cat /proc/cpuinfo`. The `flags` line should display the `avx` and `sse 4_2` flags.
- Disabled SWAP.

  You can check whether SWAP is disabled with the command `cat /proc/meminfo`. The `SwapTotal` line should display the value `0`.

  If the value is not 0:

  1. Use the command `swapon --show` to specify the device to be used as the SWAP device.
  2. In the `/etc/fstab` file, add a # sign at the beginning of the line indicating the section that you defined in the previous step.
  3. Run the command `mount -a` and verify that it completed without errors. If an error occurs, you should roll back the changes to avoid problems with loading the virtual machine.
- 300 GB of SSD disk space.
- Linux® OS:
  - Rocky Linux 9.4 and later
  - Astra Linux 1.7

**For Astra Linux**

A user account with root rights is created after installing the Astra Linux operating system. The maximum integrity level must be set for this account in order to provide the ability to install applications and write to the directory for storing application data.

Example of a command to grant the necessary privileges to a user:

```
sudo pdpl-user -i 63 username
sudo pdpl-user -i 63 root
```

kaspersky

# Licensing

The license key added to the platform defines the number of training slots for employees.

One slot is used for every user assigned training. Thus, the number of training slots used is equal to the number of users with the *Training in progress* status. Users with other statuses do not take up slots.

Additional training slots aren't required to create phishing campaigns. You can add any users to phishing campaigns (except archived users), regardless of if their training is currently in progress. However, the number of users added to phishing campaigns can't exceed the total number of available training slots.

The functionality of the test version does not differ from the full version. In the test version, five training slots are available for 2 months. This means you can add any number of users, but only five employees can take the training simultaneously on this account (not the company!).

When a commercial license key is added, the test period will expire automatically. The five training slots provided with the test version will no longer be usable.

By default, all the training slots provided with a commercial license will be available to all the companies included on the account. You can set quotas to limit a company's access to training slots.

**Adding several license keys**

The counters responsible for the maximum number of users who can be training simultaneously are considered together for all license keys valid on the current date.

For example, if you activated two license keys, one for training 2,000 users over one year, and the second for training 1,000 users over two years, then in the first year you can train up to 3,000 users simultaneously (the training slots of the two license keys are cumulative), and 1,000 users in the second year (only the second license key is valid).

**Insufficient number of training slots**

The dashboard shows how many training slots you have, how many users currently training (using these slots), and how many slots aren't currently being used. You can't start training a group that has more users than available training slots. If there aren't enough slots, you also can't manually add new users to a group where training has already started.

If there are fewer training slots than users currently training, then all training in your account will be paused. Users will still have access to the platform, but will not be able to continue training.

## In this section

kaspersky

# 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.

Please carefully read the terms of the End User License Agreement before you start using the application.

You can read the terms of the End User License Agreement in the EULA_<localization language> document located in the platform's distribution kit.

By confirming that you agree with the End User License Agreement when registering on the platform, you also accept the terms of the End User License Agreement. If you do not accept the terms of the End User License Agreement, you must cancel registration and must not use the application.

# About licensing

A *license* is a time-limited right to use the application granted under the End User License Agreement.

The license includes the right to receive the following types of services:

- using the application in accordance with the End User License Agreement
- getting technical support

The scope of services and validity period depend on the type of license under which the application was activated.

The following types of licenses are available:

- A *trial* is a free license intended for trying out the application.

  A trial license has a short term. As soon as the license expires, all Kaspersky Automated Security Awareness Platform features are disabled. To continue using the application, you need to purchase a commercial license.

  You can activate the application under the trial license only once.

- A *commercial license* is a paid license granted upon purchase of the application.

  The functions of the application stop working when a commercial license expires. To continue using the Kaspersky Automated Security Awareness Platform with full access to its functions, you must renew your commercial license.

  We recommend renewing your license no later than its expiration date to avoid any service disruptions.

# About the License Certificate

*License Certificate* – A document provided to you together with a key file or activation code.

The License Certificate contains the following information about the provided license:

- license key or order number
- information about the user to whom the license is provided

- information about the application that can be activated using the provided license
- limit on the number of license seats (for example, the number of devices on which you can use the application under the provided license)
- date of the start of the license term
- date of the end of the license term or the license term
- license type

# About the key file

The *key file* is a file with the key extension provided by Kaspersky. The purpose of a key file is to add a license key that activates the application.

The key file is emailed to the address you provided after purchasing an ASAP license or ordering a trial version of ASAP.

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

If you accidentally delete the key, it can be restored. You may need the key file, for example, to register in Kaspersky CompanyAccount.

To restore the key file, you need to do one of the following:

- Contact the license seller
- Get a key file on the Kaspersky website based on an available activation code.

# Acquiring a license

To purchase licenses, you can contact AO Kaspersky Lab partners or the company's local branches. You can find a list of partners in your region at https://locator.kaspersky.com/b2b.

Partners can also provide additional platform information and materials, information on prices, promotions, and more. A link to the search page for authorized partners is also available in the web interface of the application, in the **Licenses and companies** section.

# About data processing

All data required for Kaspersky Automated Security Awareness Platform (ASAP) is stored and processed on the side of the organization on whose server the platform is deployed. No data is transmitted to Kaspersky throughout the operation of ASAP.

During its operation, ASAP saves the following data on the device where the platform is installed:

- IDs: of employees, companies, records in the database, company administrators, companies purchasing the license, a training group of employees used for cloud synchronization, slides of training materials, and phishing campaigns.

- Data about synchronization and integration performed through external systems (SCIM, OPEN API, LOCAL AD, OUTLOOK/OWA PLUGIN (phishing-alarm)), as well as the results of this synchronization; the user's email address, appeals to users, and User data entered by the administrator.
- Data about the company training employees on the Kaspersky Automated Security Awareness Platform, including the company domain (so that all users with email addresses on this domain can be added to phishing campaigns without notifying them about the start of training) and information about the administrator.
- Data on licenses, their validity period, and the number of employees being trained.
- Data about phishing campaigns, employees, and check results, information about which emails were marked as phishing by users, and user email addresses.
- Data on the training of company employees, completed units, certificates received, and training settings.
- Internal information required for the operation of the system.

kaspersky

# Preparing to install

In this section



## Configuring the network environment

To install the platform, you must select a separate server on which no other applications will be installed.

**Create a domain for the ASAP On-Premise platform:**

1. In your organization's network, register a domain name for the platform. The domain name must be in the following format:

   `*.<domain>.<region>`

   Example: `*.kasap-domain.en`

2. Make two A-records for the IP address of your server:
   - domain

     Example: `"kasap-domain.en" in A "10.10.11.23"`
   - *.domain

     Example: `"*.kasap-domain.en" in A "10.10.11.23"`
3. Issue a wildcard SSL certificate for the platform domain with the following parameters:
   - Subject name – `*.<domain>.<region>`

     Example: `*.kasap-domain.en`
   - Alternative name – `asap-cdn.minio.<domain>.<region>`

     Example: `asap-cdn.minio.kasap-domain.en`
   - The certificate must be issued in CRT format.

4. Add the root certificates of the domain's certificate authority to the trusted certificates on the server where you want to install the platform. This can be done, for example, using the following commands:

   ```
   sudo
   cp <your_certificate_for_the_certificate_authority_of_the_main_domain>
   /etc/pki/ca-trust/source/anchors/

   sudo update-ca-trust
   ```

**Configuring rules for processing phishing domains**

During anti-phishing campaigns, users will receive emails with links to a local phishing portal, and the platform will track user transitions to it. To make this training session as believable as possible, the phishing domains must be registered on your organization's DNS server, and certificates must be issued for them:

1. On your organization's DNS servers, create a policy for users' machines, according to which A-records for the domains listed below will be resolved to an address that is local relative to the IP address where you want to deploy the ASAP On-Premise platform.
2. Issue an SSL certificate for the kasperskygroup.com domain with a list of SANs (Subject Alternative Names) for the phishing domains listed below.

   If for some reason it is impossible to issue such a shared certificate in your organization, you need to issue a separate certificate for each domain and place these certificates on the platform server in the directory with certificates for phishing domains. For example, `/etc/kasap/data/phishing-certs`.

List of phishing domains:

- www.accommodationstravel.com
- www.avviso-archiviazione.it
- www.bestjobs.solutions
- www.blockchain-info.live
- www.blog-online.live
- www.business-information.store
- www.corp-email.info
- www.correo-interno.es
- www.courrier-interne.fr
- www.docs-edit.online
- www.e-calendario.es
- www.ecalendar.ws
- www.free-clinics.co
- www.garden-club.co
- www.hotnews.agency
- www.hr-international.pro
- www.install-soft.me

- www.internal-mail.com
- www.international-lottery.club
- www.interne-mail.de
- www.justmailweb.com
- www.kaspersky.today
- www.kasperskygroup.com
- www.kreditbezahlen.de
- www.kryptomyning.com
- www.lkea.online
- www.mail-company.agency
- www.marketingservice.today
- www.medcenter.world
- www.medical-help.social
- www.netflix-cinema.com
- www.official-inbox.com
- www.official-law.site
- www.our-nature.site
- www.parties.agency
- www.paybill.email
- www.posta-interna.it
- www.postelivraison.fr
- www.share-to.me
- www.shop-delivery.store
- www.soft-exchange.com
- www.state-official.info
- www.steam-games.shop
- www.stop-covid.center
- www.storagealert.work
- www.taxpay365.com
- www.thedeliverypost.com
- www.top-programme.de
- www.travelreservation.site
- www.vosmarchandises.fr
- www.zoom-business.co

**Configure access to the SMTP mail server**

1. Make sure that the SMTP mail server is available on the configured port, for example, 587.

   An encrypted STARTTLS connection is used when connecting to the mail server (we recommend to use a version not lower than TLS 1.2)

2. For authentication on the mail server, you can use a certificate, username-password combination, or anonymous authentication. You can select the authentication method when filling out the inventory file.

   For certificate-based authentication, you must first configure the mail server accordingly and prepare a certificate in CRT format and a private key in KEY format.

**Make sure the network environment is connected**

1. Open ports 80, 443, 22, 587 on the server where you want to install the platform.
2. Make sure that port 22 is used for an SSH connection.

   When installing the platform, all ports except 80, 443, 22, and 587 are closed, and if an SSH connection is configured for a different port, it will be terminated.

# Updating Python

**Update Python**

To deploy the platform, Python version 3.9 or higher must be installed on the server. The installed version of Python can be checked using the following command:

```
python3 -V
```

Update Python as needed using the following commands:

- In the Rocky Linux operating system:

  ```
  sudo dnf update -y
  sudo dnf install python3 -y
  ```

- In the Astra Linux operating system:

  ```
  sudo apt install build-essential zlib1g-dev libncurses5-dev libgdbm-dev libnss3-dev libssl-dev libreadline-dev libffi-dev wget
  wget https://www.python.org/ftp/python/3.9.18/Python-3.9.18.tgz
  tar -zxf Python-3.9.18.tgz
  cd Python-3.9.18
  ./configure
  sudo make altinstall
  sudo update-alternatives --install /usr/bin/python3 python3 /usr/local/bin/python3.9 0
  ```

**Install Python3 pip**

Make sure pip is installed using the following command:

```
python3 -m pip --version
```

If necessary, install pip using the following commands:

- In the Rocky Linux operating system:

  ```
  sudo dnf install python3-pip -y
  ```

- In the Astra Linux operating system:

  ```
  sudo apt install python3-pip
  ```

**Install Python3 pip dependencies**

Install the required Python3 pip dependencies using the following command:

```
sudo python3 -m pip install ansible kubernetes pyyaml openshift
```

# Preparing the platform cluster

In the fault-tolerat version, the ASAP On-Premise platform is deployed in a cluster of multiple duplicate servers, or "target machines." Another server, the "control machine," is used for installation and management.

**Create a user for platform deployment with the required rights**

To deploy the platform, you need to create a user with sudo rights on the control machine and all target machines.

> When deploying on Astra Linux, issue the maximum `sudo pdpl-user -i 63` integrity level on all cluster servers.

**Provide SSH access from the control machine to the target machines of the platform cluster**

► *To provide SSH access from the control machine to the target machines of the platform cluster:*

> The steps below must be performed on behalf of the user who will deploy the platform.

1. Log in to the monitoring machine and change to the `~/.ssh directory`. If this directory doesn't exist, it must be created.
2. Create an SSH key using the `ssh-keygen -t rsa` command.

   While executing the command, answer all system questions by pressing the **ENTER** key. As a result of the command execution, the key file `~/.ssh/id_rsa.pub` should appear on the control machine.
3. Copy the contents of the `~/.ssh/id_rsa.pub` file.
4. Log in to the target machine.
5. Create a `~/.ssh/authorized_keys` file and place in it the contents of the `id_rsa.pub` key that was previously created on the control machine.

6. Repeat steps 4-5 for each target machine in the platform cluster.
7. Log in to the monitoring machine and make an SSH request to each target machine in the cluster.
8. Confirm your consent to the key exchange. This request appears the first time you connect to the target machine.

SSH access from the control machine to the target machines is configured. Requests are executed without prompting for a password and without errors.

# Synchronizing time on servers

Time synchronization must be configured for all machines in the cluster.

**For the Astra Linux operating system**

► *To configure time synchronization on servers:*

1. Make sure the virtual machine has Internet access.

   If access is available, go to step 2.

   If there's no access, edit the `/etc/systemd/timesyncd.conf` file and add the `NTP=<Address of the NTP server in the organization>` parameter in the `[Time]` section. Example:

   ```
   [Time]
   NTP=10.10.10.1
   ```

2. Start the system time synchronization service by executing the following commands:

   ```
   sudo systemctl enable systemd-timesyncd.service
   sudo systemctl start systemd-timesyncd.service
   ```

3. Wait a few seconds and run the following command:

   ```
   timedatectl
   ```

   If the system time is synchronized correctly, the output will contain the following fields:

   NTP service: active

   System clock synchronized: yes

Synchronization is configured.

**For the Rocky Linux operating system**

► *To configure time synchronization on servers:*

1. Install chrony:

   ```
   sudo apt install chrony
   ```

2. Configure the system time to synchronize with the NTP server:

   a. Make sure the virtual machine has Internet access.

      If access is available, go to step b.

      If internet access is not available, edit the `/etc/chrony.conf` file to replace `2.pool.ntp.org` with the name or IP address of your organization's internal NTP server.

   b. Start the system time synchronization service by executing the following command:

      ```
      sudo systemctl enable --now chronyd
      ```

   c. Wait a few seconds and run the following command:

      ```
      sudo timedatectl | grep 'System clock synchronized'
      ```

      If the system time is synchronized correctly, the output will contain the line "System clock synchronized: yes".

Synchronization is configured.

kaspersky

# Test machine preparation

When deploying a fault-tolerant platform, the installer and inventory files are located on the control machine.

## In this section



## Unpacking the distribution package

The files need to be moved, created and edited with sudo privileges.

► *Prepare the directory structure for the installer:*

1. Create a directory for the platform installation files. For example, `/etc/kasap/distributive`:

   ```
   sudo mkdir /etc/kasap/distributive -p
   ```

2. Create a directory for inventory files. For example, `/etc/kasap/data`:

   ```
   sudo mkdir /etc/kasap/data -p
   ```

3. Unpack the archive for installing the ASAP On-Premise components (see the "Distribution kit" section on p. 4) into the directory for platform installation files (for example, `/etc/kasap/distributive`).

## Creating inventory files

In the directory created for inventory files (see the "Unpacking the distribution package" section on p. 18), create two files with the platform deployment settings:

- **user_variables.yml** – in this file, list the platform deployment parameters (see the "Structure of the user_variables.yml file" section on p. 32). See below for an example of the contents of the file.
- **inventory.ini** - in this file, specify the hosts of the control machine and the target machines. See below for an example of the contents of the file.

  In the fail-safe version, the platform is deployed on one control machine and several target machines.

  - Target machines duplicate each other to ensure the platform is operational even if any of the servers are disconnected.
  - The control machine is responsible for data synchronization between target machines and the control machine. The platform is also deployed from it.

kaspersky

### Example of the contents of user_variables.yml

```
base_domain: "company.local"


ingress_ips:
  - 10.10.10.10


main_certification_authority_crt_path: "/etc/kasap/data/ca.crt"
smtp_certification_authority_crt_path: "/etc/kasap/data/smtp-ca.crt"


application_certificate_crt_path: "/etc/kasap/data/main.crt"
application_certificate_key_path: "/etc/kasap/data/main.key"


application_locales:
"en,bs,cs,ca,da,es,mx,de,fr,hy,hr,it,hu,nl,pl,br,pt,ro,sv,tr,el,ru,kk,sk,sr,
ar,ja,cn,zh"


application_smtp_port: 587
application_smtp_host: <smtp_server_host>
application_smtp_auth_type: credentials
application_smtp_secure: false
application_smtp_require_tls: true
application_smtp_user: smtp-user
application_smtp_password: "...."


application_phishing_certificates_folder: "/etc/kasap/data/phishing-certs/"
```

English is required to install Kaspersky ASAP On-Premise. That means one of the values for the `application_locales` parameter must be `en`. For example, `application_locales: "en"`.

kaspersky

### Example of the contents of inventory.ini

```
[control]
control ansible_host=<internal_address_of_host> ansible_connection=local

[k3s]
server-1 ansible_host=<internal_host_address_1>
server-2 ansible_host=<internal_host_address_2>
server-3 ansible_host=<internal_host_address_3>

[all:vars]
ansible_user=<service_account_name>
ansible_ssh_common_args=<ssh_connection_parameters>
ansible_ssh_private_key_file=<path_to_private_key_service_account>
```

kaspersky

# Installing ASAP On-Premise

In this section



## Installing the platform

► *To install the ASAP On-Premise platform:*

1. Make sure that the machine on which you are going to install Kaspersky ASAP On-Premise meets the hardware and software requirements (see the "Hardware and software requirements" section on p. 5), and also that you have completed the preparatory steps for installation (see the "Preparing for installation" section on p. 11).

   If you're installing the platform on top of an already deployed Kaspersky ASAP On-Premise platform (for example, to update the version (see the "Migration from version 1.0 to 1.1" section on pg. 27) or certificates), all languages used in the deployed version of the platform must be specified in the inventory file.

2. Go to the directory where you unpacked the platform distribution package, and go to the `installer` subdirectory. For example, `/etc/kasap/distributive/installer`.

3. Run the following command:

   ```
   sudo ./run.sh install.yml -i -data-path=<directory with inventory files>
   ```

   For example:

   ```
   sudo ./run.sh install.yml -i -data-path=/etc/kasap/data
   ```

4. Before starting the installation, you will be prompted to review the terms of the End User License Agreement (see the "About the End User License Agreement" section on p. 8) and accept them. Review and accept the End User License Agreement.

   If you do not accept the terms of the End User License Agreement, the Kaspersky ASAP On-Premise platform will not be installed.

The Kaspersky ASAP On-Premise platform is installed.

We recommend that you save the inventory files used to install the platform. They can be used to change the platform settings or uninstall Kaspersky ASAP On-Premise.

kaspersky

# Installation result

When installation is complete, the following services will be created:

- https://asap-api.<domain>.<region> (for example: `https://asap-api.kasap-domain.en`) – used to integrate the platform with other solutions via the API.
- https://app.<domain>.<region> (for example: `https://app.kasap-domain.en`) – used to log in to the platform's web interface.
- https://*.<domain>.<region>, for example: `https://*.kasap-domain.en`
- https://cdn.<domain>.<region>, for example: `https://cdn.kasap-domain.en`
- https://test-player.<domain>.<region>, for example: `https://test-player.kasap-domain.en`
- https://minio.<domain>.<region>, for example: `https://minio.kasap-domain.en`
- https://minio-console.<domain>.<region>, for example: `https://minio-console.kasap-domain.en`
- https://asap-cdn.minio.<domain>.<region>, for example: `https://asap-cdn.minio.kasap-domain.en`

kaspersky

# Verifying the installation result

► *To verify that the ASAP On-Premise platform is installed correctly:*

- Go to the platform's URL for logging in (use a URL in the following format: https://app.<domain>.<region>, for example, `https://app.kasap-domain.en`) and verify that the application is available: the login window should be displayed and prompt you to enter your login and password.
- Go to any of the previously configured phishing domains and make sure that a 404 page is displayed for it (this is the correct behavior). Also, in the browser, in the page connection settings in the Network section, verify that the response to the /server-list.json request contains the URL in the https://asap-api.<domain>.<region> format.

If both conditions are met, then the installation is correct.

► *To verify the operability of the ASAP On-Premise platform, the platform administrator must:*

1. Go to the platform's URL for logging in (use a URL in the following format: https://app.<domain>.<region>, for example, `https://app.kasap-domain.en`).

   The platform's login window opens.
2. Click the **Register** link and complete the registration process by specifying your email address and password.
3. Wait for the registration confirmation email and click the link in it.

   After registration is complete, the page with the dashboard should be displayed.
4. Go to the **Contents** page, open any lesson in the **Lesson** section, and make sure that the test player displays content.

If you were able to complete all the steps, the platform was installed correctly.

# Removing ASAP On-Premise

► *To remove the ASAP On-Premise platform:*

1. Go to the directory where you unpacked the platform distribution package, and go to the `installer` subdirectory. For example, `/etc/kasap/distributive/installer`.
2. Run the following command:

   ```
   sudo ./run.sh uninstall.yml -i -data-path=<directory with inventory files>
   ```

   For example:

   ```
   sudo ./run.sh uninstall.yml -i -data-path=/etc/kasap/data
   ```

The Kaspersky ASAP On-Premise platform is removed.

kaspersky

# About a backup copy

This section is about working with backup copies of the ASAP On-Premise platform: how to create and deploy them.

## In this section



## Creating a backup copy

► *To create a backup copy of platform components:*

1. Unpack the archive from the platform distribution kit (see the "Distribution kit" section on p. 4) and go to the `installer` subdirectory. For example, `/etc/kasap/distributive/installer`.
2. Run the following command:

   ```
   sudo ./run.sh backup.yml -i -data-path=<directory with inventory files>
   ```

   For example:

   ```
   sudo ./run.sh backup.yml -i -data-path=/etc/kasap/data
   ```

The selected components will be backed up.

# Deploying from a backup copy

► *To deploy a component from a previously created backup copy:*

1. Unpack the archive from the platform distribution kit (see the "Distribution kit" section on p. 4) and go to the `installer` subdirectory. For example, `/etc/kasap/distributive/installer`.
2. Run the following command:

   ```
   sudo ./run.sh restore.yml -i -data-path=<directory with inventory files> -extra-vars=<directory with backup copy>
   ```

   For example:

   ```
   sudo ./run.sh restore.yml -i -data-path=/etc/kasap/data -extra-vars=/etc/backup
   ```

The components will be deployed from the backup copies.

# Migration from version 1.0 to version 1.1

► *To migrate from ASAP On-Premise version 1.0 to version 1.1:*

1. Back up the ASAP On-Premise version 1.0 platform components.
2. Install ASAP On-Premise Platform version 1.1 (see Installing ASAP On-Premise on page 21). Make sure that all languages installed in version 1.0 of the platform are selected when installing version 1.1.
3. Place the backup copies of ASAP On-Premise version 1.0 in the following directories:

   ```
   <Directory for inventory files>/backup /minio/<directory with MinIO backup>
   ```

   ```
   <Directory for inventory files>/backup/mongo/<directory with Mongo backup>
   ```

   Example:

   ```
   /etc/kasap/data/backup/minio/minio_backup
   ```

   ```
   /etc/kasap/data/backup/mongo/mongo_backup
   ```

4. Go to the directory where you unpacked the archive from the platform distribution kit (see "Distribution kit" on page 4) and open the `installer` subdirectory. For example, `/etc/kasap/distributive/installer`.
5. Run the following command:

   ```
   sudo ./run.sh migrate_v1_to_v2.yml -i -data-path=<directory with inventory files> -extra-vars="minio_backup_date=<directory with MinIO backup> mongo_backup_date=<directory with Mongo backup>"
   ```

   Example:

   ```
   sudo ./run.sh migrate_v1_to_v2.yml -i -data-path=/etc/kasap/data -extra-
   vars="minio_backup_date=/etc/kasap/data/backup/minio/minio_backup
   mongo_backup_date=/etc/kasap/data/backup/mongo/mongo_backup"
   ```

Migration of ASAP On-Premise in version 1.0 to version 1.1 is complete.

# Receiving the platform activity log

► *To create an ASAP On-Premise activity log:*

1. Go to the `installer` subdirectory in the installer directory. For example, `/etc/kasap/distributive/installer`.
2. Run the following command:

   ```
   sudo ./run.sh logs.yml -i -data-path=<directory with inventory files>
   ```

   For example:

   ```
   sudo ./run.sh logs.yml -i -data-path=/etc/kasap/data
   ```

In the directory with inventory files, the `logs` directory was created (for example, `/etc/kasap/data/logs)`, where the archive with the platform activity log will be placed.

# Sources of information about the application

### The Kaspersky Automated Security Awareness Platform page on the Kaspersky website

On the Kaspersky Automated Security Awareness Platform page (https://www.kaspersky.com/small-to-medium-business-security/security-awareness-platform), you can view general information about the application, its functions, and its features.

The Kaspersky Automated Security Awareness Platform page contains a link to the online store. This is where you can purchase or renew the application.

### Discussing Kaspersky applications on the Forum

If your question does not require an immediate answer, you can discuss it with Kaspersky experts and other users on our Forum (https://forum.kaspersky.com/forum/english-forum-161/).

Here you can view existing topics, leave your comments, or create new topics.

# Information about third-party code

Information about third-party code is contained in the LEGAL_NOTICES file located in the platform's distribution kit.

# Trademark notices

Registered trademarks and service marks are the property of their respective owners.

Google Chrome is a trademark of Google LLC.

Linux is the trademark of Linus Torvalds, registered in the United States and in other countries.

Python is a trademark or registered trademark of the Python Software Foundation.

kaspersky

# Applications

In this section



## Structure of the user_variables.yml file

| Variable | Type | Description | Example |
|---|---|---|---|
| `base_domain` | string | The base domain on which you want to host the platform. | `base_domain: "company.local"` |
| `ingress_ips` | array of IP addresses | An array of IP addresses for which the platform's web resources will be available. | `ingress_ips:`<br>`- 10.10.10.10`<br>`- 192.168.0.1` |
| `main_certification_authority_crt_path` | string, path | Path to the Certification Authority from the platform certificate. | `main_certification_authority_crt_path: "/etc/kasap/data/ca.crt"` |
| `application_certificate_crt_path` | string, path | Path to the public part of the platform certificate. | `application_certificate_crt_path: "/etc/kasap/data/main.crt"` |
| `application_certificate_key_path` | string, path | Path to the private part of the platform certificate. | `application_certificate_key_path: "/etc/kasap/data/main.key"` |

| Variable | Type | Description | Example |
|---|---|---|---|
| application_phishing_certificates_folder | string, path | Path to the folder with phishing domain certificates. The public parts with the .crt extension and the private parts with the .key extension need to be put inside.<br>The name of the certificate and its key must match. | application_phishing_certificates_folder "/etc/kasap/data/certs/" |
| application_locales | string | Comma-separated list of language codes for the languages in which you want to deploy the platform (see the "Available languages and corresponding codes" section on p. 35). | application_locales: "en,zh" |
| application_smtp_port | number | Port for connecting to the SMTP server. | application_smtp_port: 587 |
| application_smtp_host | string | Host for connecting to the SMTP server. | application_smtp_host: smtp.company.local |
| smtp_certification_authority_crt_path | string, path | Path to the Certification Authority from the SMTP host. | smtp_certification_authority_crt_path: "/etc/kasap/data/smtp.ca" |
| application_smtp_auth_type | Available values:<br>• anonymous<br>• credentials<br>• certificate | Authorization type on the SMTP server. | application_smtp_auth_type: credentials |
| application_smtp_secure | Available values:<br>• true<br>• false | Whether to use SMTPS for connecting to the SMTP server. | application_smtp_secure: false |

| Variable | Type | Description | Example |
|---|---|---|---|
| application_smtp_require_tls | Available values:<br>• true<br>• false | Whether to use SSL/TLS for connecting to the SMTP server. | application_smtp_require_tls: true |
| application_smtp_user | string | User for connecting to the SMTP server. Required if the application_smtp_auth_type is set to "credentials". | application_smtp_user: user@example.com |
| application_smtp_password | string | Password for connecting to the SMTP server. Required if the application_smtp_auth_type is set to "credentials".<br>The password isn't required. In this case, you'll need to enter it later during the installation process. | application_smtp_password: password |
| application_smtp_public_key_path | string, path | Path to the public part of the client certificate for connecting to the SMTP server. Required if the application_smtp_auth_type is set to "certificate". | application_smtp_public_key_path: "/etc/kasap/data/smtp.crt" |
| application_smtp_private_key_path | string, path | Path to the private part of the client certificate for connecting to the SMTP server. Required if the application_smtp_auth_type is set to "certificate". | application_smtp_private_key_path: "/etc/kasap/data/smtp.key" |

kaspersky

# Available languages and corresponding codes:

- Bosanski – bs
- Català – ca
- Dansk – da
- Deutsch – de
- English – en (language required for installation)
- Español (España) – es
- Français – fr
- Hrvatski – hr
- Italiano – it
- Magyar – hu
- Nederlands – nl
- Polski – pl
- Português (Brasil) – br
- Português (Portugal) – pt
- Română – ro
- Slovenski – sk
- Srpski – sr
- Svenska – sv
- Türkçe – tr
- Čeština – cs
- Ελληνικά – el
- Español (México) – mx
- Русский – ru
- Қазақша – kk
- Հայերեն – hy
- العربية – ar
- 日本語 – ja
- 繁體中文 – cn
- 简体中文 – zh