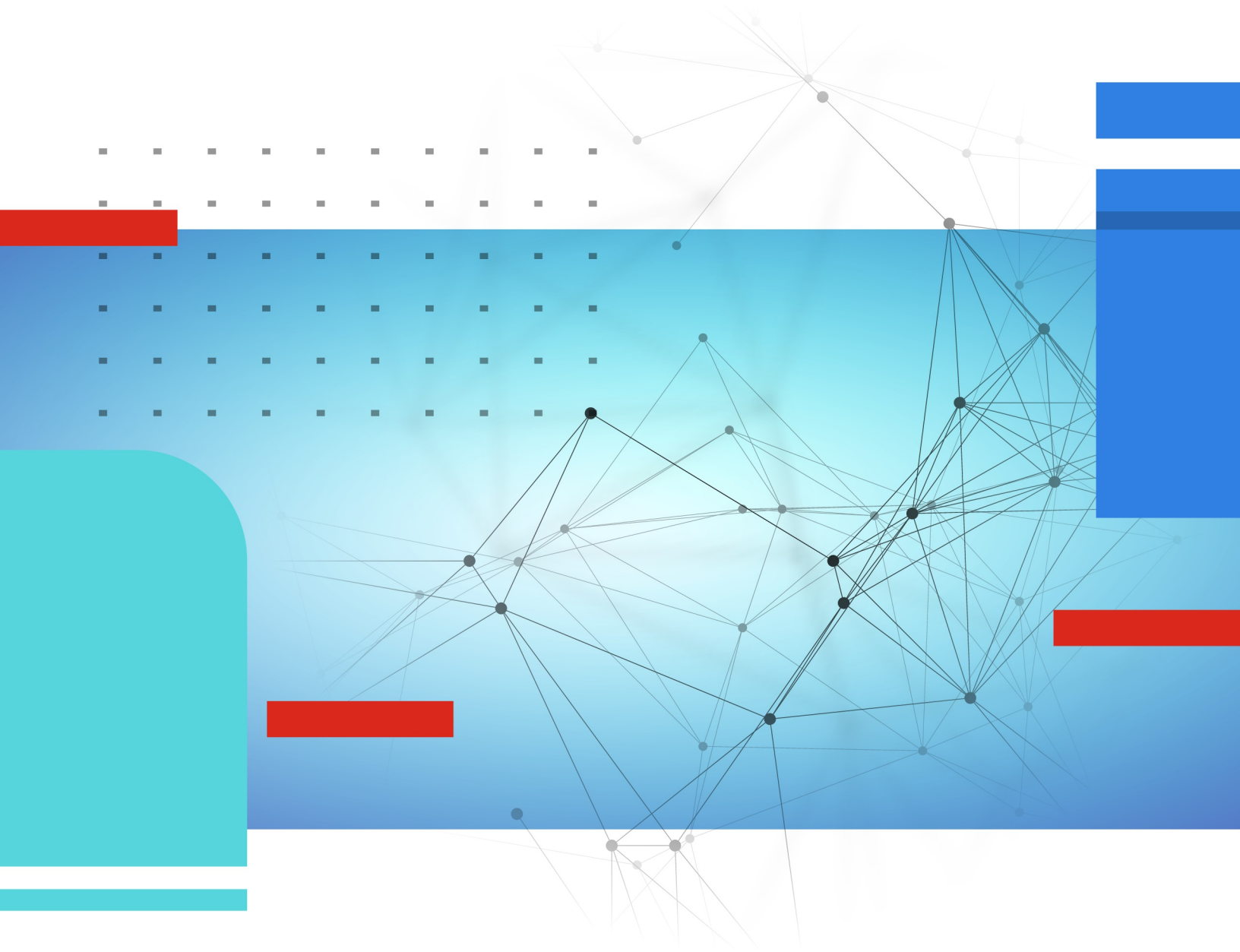
Getting Started (Docker)

Container FortiOS 7.2.1



FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO LIBRARY
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/training-certification

FORTINET TRAINING INSTITUTE
https://training.fortinet.com

FORTIGUARD LABS
https://www.fortiguard.com

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

June 13, 2025
Container FortiOS 7.2.1 Getting Started (Docker)
87-721-971807-20250613

TABLE OF CONTENTS

Change Log

Date	Change Description
2024-07-17	Initial release.
2024-07-24	Updated Creating Docker networks on page 7.
2024-11-08	Updated Uploading a license.
2025-06-13	Updated Getting the container image on page 6.

Introduction

Container FortiOS provides NGFW firewall features, including security policies, IPS inspection, application control, URL filtering, and antivirus in a container-deployed format.

It supports Linux Containers (LXC), Docker, and Kubernetes.

This guide provides information about the installation and configuration of Container FortiOS version 7.2.1, build 0255 on Docker.

Deploying Container FortiOS

This section provides an overview of the procedures for deployment of Container FortiOS.

As container environments vary widely, this document provides basic instructions for deployment to Docker and does not provide in-depth information about configuration of Docker itself.

The basic steps for deployment are as follows:

1. Get the container image.
2. Import the image.
3. Create networks.
4. Specify the persistent storage location.
5. Optionally, specify startup configuration.
6. Create the container.
7. Start the container.
8. Connect to networks.
9. Upload a valid license.

Getting the container image

After purchasing a Container FortiOS license, submit a ticket through the Customer Service & Support site. The Technical Assistance center (TAC) team will then provide you with the appropriate image file.

For more information about submitting a ticket, see Technical Tip: How to create a ticket for Fortinet TAC.

Docker images use the following naming convention:

```
FOS_<CPU Arch>_<Container Type>-v<Major Version>-build<build number>-<Company>.tar.gz
```

For example, image `FOS_X64_DOCKER-v7-build0255-FORTINET.tar.gz` was built for Docker running on an 64 bit Intel CPU device. The major version is 7 and build number is 0255.

Loading the container image

Import the Docker image tarball using the following command:

```
sudo docker load -i FOS_X64_DOCKER-v7-build0255-FORTINET.tar.gz
```

Check that the image has been imported and tagged with the following command:

```
sudo docker images
```

Creating Docker networks

You will need to configure the required networks for your case and attach them to the Container FortiOS container. Any supported network interfaces can be used.

Docker creates a default bridge network for containers, `docker0`, that can be used for egress traffic.

For more information about the Docker bridge driver, see https://docs.docker.com/network/drivers/bridge.

To create a MACVLAN network:

The following commands show examples of creating MACVLAN networks:

```
sudo docker network create --driver=macvlan \
--subnet=10.210.16.0/24 \
--gateway=10.210.16.1 \
-o parent=enp0s31f6 \
00-cFOS-WAN
```

```
sudo docker network create --driver=macvlan \
--subnet=192.168.254.0/24 \
--gateway=192.168.254.1 \
--aux-address="rtos=192.168.254.1" \
--aux-address="this-host=192.168.254.2" \
-o parent=enp0s3f1 \
99-cFOS-LAN
```

This example creates two MACVLAN networks, a WAN network named `00-cFOS-WAN` and a LAN network named a WAN network named `99-cFOS-LAN`, each bridging one of the host NICs.

`aux-address` instructs Docker to exclude the specified IP address from being used in the defined MACVLAN network, such as when a given IP address is already in use.

Use the MACVLAN network driver to assign a MAC address to your container virtual network interface, making it appear to be a physical network interface directly connected to the physical network. In this case, you need to designate a physical interface on your Docker host to use for the MACVLAN, as well as the subnet and gateway of the network.

For more information about the Docker MACVLAN network driver, see https://docs.docker.com/network/drivers/macvlan/.

Specifying persistent storage

Container FortiOS containers require persistent local storage. For each container, specify a local directory on the host.

Specifying startup configuration

You may specify a license and configuration to be applied when the container is created. The license must be provided for the configuration to be applied.

You may specify either a full configuration file or a partial configuration file, but not both.

Full configuration file

A full backup configuration file can be exported from a running container:

```
exec config backup <file name> [password]
```

The full configuration file must be encrypted with a password in order to be applied to a container with a different data directory.

Partial configuration file

A partial configuration file can be applied at startup, but it cannot be encrypted.

Specifying configuration and license at startup

To specify configuration and license at startup:

1. Copy the backup file to the `data` directory and name it `cfos.conf`.

   When using an encrypted configuration file at startup, save the backup password in a file named `cfos.key` in the data directory.

   You may also restore a partial configuration using a `cfos-partial.conf` file.
2. Copy the Container FortiOS license file to the `data` directory and name it `cfos.lic`.

   The license and configuration files are read and applied to the container when it runs, then deleted automatically.

Creating the container

Use the `docker container create` command to create the Container FortiOS container and prepare it to be run:

```
sudo docker container create \
--network 00-cFOS-WAN \
--ip=10.210.16.254 \
-p 192.168.11.248:2431:2431 \
-p 192.168.11.248:2432:2432 \
-p 192.168.11.248:4022:4022 \
-p 192.168.11.248:5443:5443 \
-p 192.168.11.248:8080:8080 \
-p 192.168.11.248:500:500/udp \
-p 192.168.11.248:4500:4500/udp \
--cap-add=NET_ADMIN \
```

```
--cap-add=SYS_ADMIN \
--security-opt apparmor:unconfined \
--name cfos1 \
-v/srv/cfos/cfos1_data:/data \
--dns 96.45.45.45 \
--dns 96.45.46.46 \
-it fos
```

Replace the values in the command with appropriate values for your configuration. In this example, `192.168.11.248` is the host external IP address.

For more information about Docker options for this command, see the Docker documentation for the docker create command.

Starting the container

Start the Docker container with the following command:

```
sudo docker container start --attach -i cfos1
```

This command starts the container created in the previous step, named `cfos1`, and attaches to its pseudo-TTY.

Connecting to networks

To connect the running container to the networks you defined, use the `docker network connect` command, as shown in the following example:

```
sudo docker network connect --ip 192.168.254.254 99-cFOS-LAN cfos1
```

Using Container FortiOS

This section provides an overview of the initial steps for connecting to and configuring the running Container FortiOS container.

Connecting to the Container FortiOS CLI

Container FortiOS provides access to the FortiOS shell for CLI usage as well as the underlying Linux shell.

The Container FortiOS CLI is based on the FortiOS CLI, but has fewer available options.

To connect to the running Container FortiOS container:

In the host shell, enter the following command:

```
sudo docker exec -it <container_name> /bin/cli
```

The initial username is `admin` with an empty password. Use `config system admin` to set a password.

To enter the Linux shell, use the following command:

```
sysctl sh
```

Connecting to the REST API

Container FortiOS provides a REST API to perform configuration and monitoring operations. The API provides a subset of the FortiOS API.

The API is accessible by default on port 443 at any of the container interfaces. If configured to require a token, all requests must include the API token.

For example, the following examples get the antivirus settings:

```
curl -H "Authorization: Bearer rkMJd3SdLhb8UFBan987CnIrmPBLfaIj"
https://localhost/api/v2/cmdb/antivirus/settings
```

```
curl https://localhost/api/v2/cmdb/antivirus/settings?access_
token=rkMJd3SdLhb8UFBan987CnIrmPBLfaIj
```

To see the available options for each object, append `?action=schema` to the url.

For full details on the available API actions and access, see the Container FortiOS REST API documentation on FNDN.

Uploading a license

You must upload a valid Container FortiOS license before many features are available.

To upload a license:

1. Open the license file in a text editor and copy the full contents.
2. In the container CLI, enter the following command:

   ```
   exec import-license "<content_of_license_file>"
   ```

   Container FortiOS validates the license.

More information

Additional Container FortiOS documentation is available in the Fortinet Documentation Library.

FortiOS documentation

Configuration and administration of Container FortiOS is very similar to FortiOS.

The following FortiOS documentation may be helpful:

- FortiOS Administration Guide
- FortiOS CLI Reference
- FortiOS REST API Reference on FNDN

www.fortinet.com