



# Kaspersky SD-WAN

## Proof of Concept

Part 1: deployment, configuration & traffic management

15.05.2025

Document version:  
2.4.0.0

RGB

0 168 142	29 29 27	51 194 255	51 92 255	112 51 255	255 51 92	239 237 238	239 255 252
-----------	----------	------------	-----------	------------	-----------	-------------	-------------

## Change log

Date	Info
26.09.2023	Initial version.
03.10.2023	Update for release 2.1.1.
22.04.2024	Update for release 2.2.0.
15.05.2024	Update for release 2.2.1.
13.08.2024	Revised text.
06.12.2024	Update for release 2.3.1.
15.05.2025	Update for release 2.4.0.

## Contents

<b>1. Kaspersky SD-WAN</b>	<b>6</b>
1.1. Kaspersky SD-WAN Architecture	7
<b>2. Description of the Kaspersky SD-WAN PoC</b>	<b>8</b>
2.1. Kaspersky SD-WAN PoC topology	9
2.2. PoC IP address plan and resource requirements for SD-WAN components	10
2.3. Network ports used by core system components	12
2.4. SD-WAN containers' external connections diagram	13
2.5. Software versions	14
2.6. Hardware requirements for the Kaspersky SD-WAN solution	14
<b>3. Kaspersky SD-WAN solution components: installation and configuration</b>	<b>15</b>
3.1. Operation system installation on orc1 host	15
3.2. Installation of Kaspersky SD-WAN Management system components	27
3.3. Connecting to the Kaspersky SD-WAN management console	34
3.4. Logging in to the Zabbix monitoring system and set up configuration	36
<b>4. Basic Kaspersky SD-WAN configuration</b>	<b>39</b>
4.1. Creating domains and data centers	39
4.2. SD-WAN instance template configuration	48
4.3. Creating an SD-WAN service template	51
4.4. Creating a tenant and deploying an SD-WAN service	56
4.5. Creating firewall template for SD-WAN gateways	65
4.6. Creating CPE templates for SD-WAN gateways	69
4.7. Importing CA certificates for CPE devices	89
4.8. SD-WAN gateways initial configuration	91
4.9. SD-WAN gateways registration	94
4.10. Configuring the Management P2M Service	102
4.11. CPEs initial configuration	106
4.12. Creating firewall template for CPE devices	109
4.13. Creating CPE devices templates	112
4.14. CPE device registration	128
<b>5. Traffic management</b>	<b>135</b>
5.1. Creating a Layer 2 Multipoint-to-Multipoint (M2M) transport service	135
<b>6. Verifying BGP and VRRP protocols on CPEs</b>	<b>139</b>
<b>7. Upgrading the components of the Kaspersky SD-WAN Management System</b>	<b>144</b>
<b>Appendix A. Configuration of infrastructure components</b>	<b>145</b>
ISP router	145



R13 router .....	151
R14 router.....	153
R11 router.....	155
R12 router .....	156
<b>Appendix B. Acceptance test plan .....</b>	<b>157</b>

# 1. Kaspersky SD-WAN

A software-defined wide area network (SD-WAN) is a wide area network that uses software-defined network technology, such as communicating over the Internet using encrypted overlay tunnels for distributed company networks.

A key application of SD-WAN is to allow companies to build higher-performance WANs using lower-cost and commercially available Internet access, enabling them to partially or wholly replace more expensive private WAN connection technologies such as MPLS.

Kaspersky SD-WAN addresses the key shortcomings of the existing traditional WAN networks. The Kaspersky SD-WAN solution is a replacement for traditional networking approach, standard WAN routers, provides predictable, optimized access to business-critical applications, is agnostic to WAN transport technologies, and can use any WAN links.

The Kaspersky SD-WAN solution allows you to build a reliable, geographically distributed, and fast-time scalable corporate network with application-aware efficient routing and simplified centralized management.

Kaspersky SD-WAN combines the following key features:

- Centralized, on-prem or cloud-delivered management, multi-tenancy, and Role Based Access Control (RBAC)
- Template-based Zero Touch Provisioning (ZTP) to speed up the connection of new company sites and remove human error
- High Availability with prioritization of critical business-applications.
- Load-balancing via multiple WAN links
- Full mesh and partial mesh topologies
- Network security functions deployment as Virtualized Network Functions (VNFs) and integration into user traffic chains
- Intelligent traffic management

## 1.1. Kaspersky SD-WAN Architecture

Description of the main components of Kaspersky SD-WAN:

- SD-WAN orchestrator provides a graphical management interface, CPE device inventory, configuration templates, transport service policies, and CPE device registration
- SD-WAN controller manages overlay network, builds transport services, performs tunnel quality monitoring, automatically switches application traffic to backup channels, and performs CPE management via the OpenFlow protocol
- SD-WAN gateways are CPE devices with an assigned SD-WAN gateway topology role. Terminates overlay tunnels from CPE devices and forms an SDN fabric in the form of an overlay network
- Kaspersky Edge Service Routers (KESR) / Customer Premise Equipment (CPE) - telecommunications equipment that connects to SD-WAN gateways and other CPE devices using overlay tunnels and forward data traffic

The architecture of Kaspersky SD-WAN is presented in Figure 1.

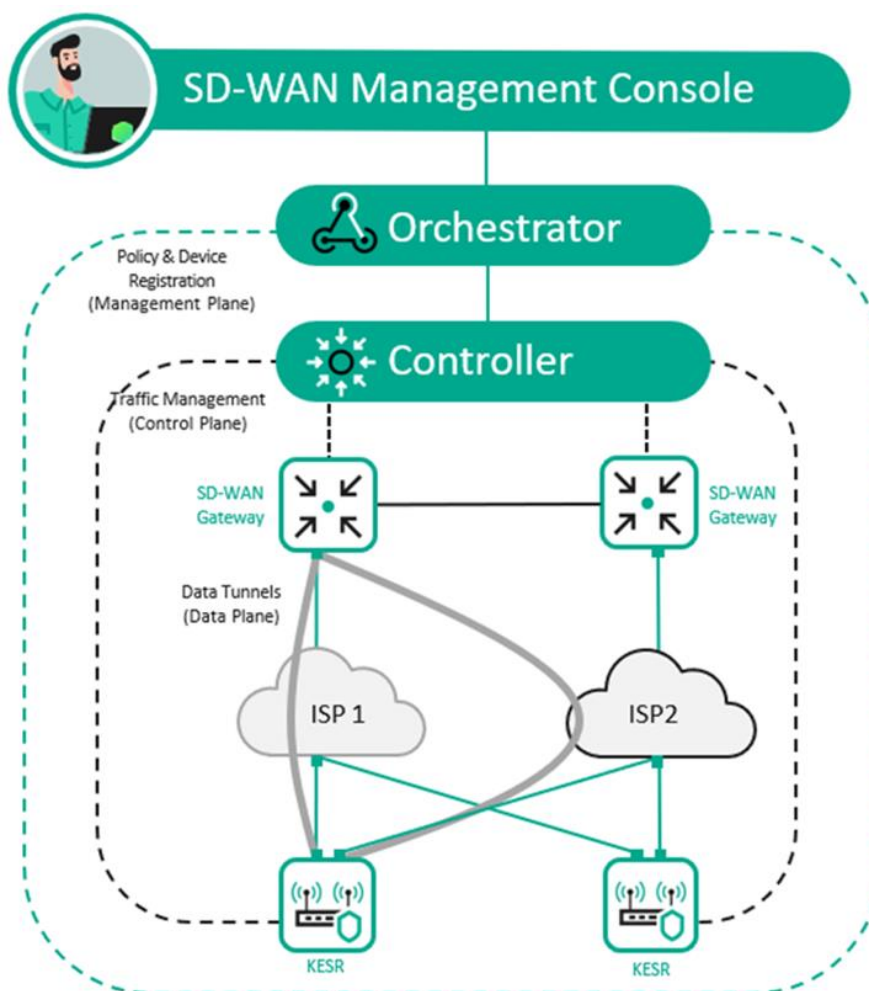


Figure 1 - Kaspersky SD-WAN architecture.

## 2. Description of the Kaspersky SD-WAN PoC

The Kaspersky SD-WAN PoC components are deployed in a VMWare virtualization environment.

The Kaspersky SD-WAN management components such as SD-WAN orchestrator, controller, and monitoring system are deployed as Docker containers on the orc1 virtual host.

Figure 2 shows the topology of the Kaspersky SD-WAN PoC.

PoC topology description:

- The DC site has two network segments, dc-lan1 and oob, that are connected to router R13. The orc1 virtual machine is connected to the oob segment, while the srv1 server is connected to the dc-lan1 segment and hosts the test WWW service
- There are two routers, R11 and R12, at the DC boundary. Behind them are two SD-WAN gateways: vGW-11 and vGW-12. The internal (lan) interfaces of R13, vGW-11, and vGW-12 are connected to the dc-perim network segment
- Routers R11 and R12 perform Source Network Address Translation (SNAT) for vGW-11 and vGW-12 and Destination Network Address Translation (DNAT) for the ports specified in Table 1 for connecting CPE devices
- Router R14 carries out SNAT and acts as the default gateway for Router R13. Additionally, it serves as an Internet gateway for the host orc1. Router R14 provides DNAT for the SD-WAN orchestrator and SD-WAN controller, the ports are specified in Table 1
- The ISP host emulates the connection to the Internet with ISP1 - ISP8 service providers.
- SD-WAN gateways must be accessible through the network ports listed in Table 2.
- The vCPE-3 device is an example of a remote site with one CPE device connected to two carriers
- The vCPE-4 device represents a future scenario in which a remote site is connected using a universal uCPE device that is not currently part of this PoC
- The vCPE-51 and vCPE-52 gateways are an example of High-Availability scenario. The CPE devices support the VRRP protocol

## 2.1. Kaspersky SD-WAN PoC topology

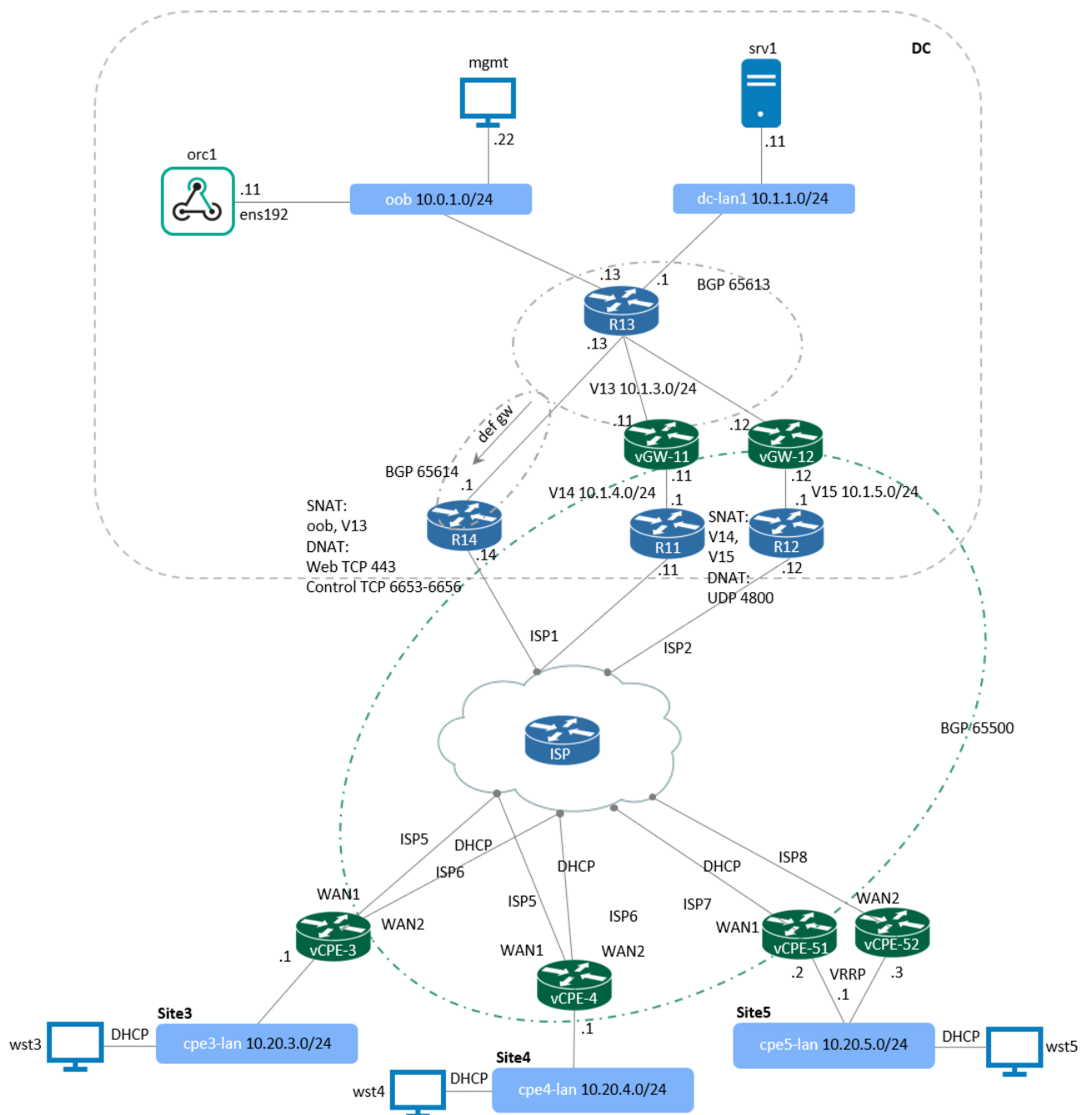


Figure 2 - Kaspersky SD-WAN PoC topology.

## 2.2. PoC IP address plan and resource requirements for SD-WAN components

This IP plan corresponds to the scheme in paragraph 2.1. If other addresses are used, it is necessary to change the IP plan and all SD-WAN settings in further steps.

Table 1 - Host parameters used in the PoC.

Host	Operation system	IP address	Description	System requirements
orc1	Ubuntu 22.04.06 LTS Server	10.0.1.11	Docker containers: www-1, orc-1, redis-1m, mongo-1, vnfm-1, vnfm- proxy-1, ctl-1, zabbix-www-1, zabbix-srv-1, zabbix-proxy-1, zabbix-db-1, syslog-1, mockpnf-1	24 x vCPU, 24 GB RAM
vGW-11	vKESR-M2 image	wan 10.1.4.11 lan 10.1.3.11 overlay 172.16.1.11	SD-WAN gateway	4 x vCPU, 8 GB RAM
vGW-12	vKESR-M2 image	wan 10.1.5.12 lan 10.1.3.12 overlay 172.16.1.12	SD-WAN gateway	4 x vCPU, 8 GB RAM
vCPE-3	vKESR-M1 image	wan DHCP lan 10.20.3.1 overlay 172.16.1.3	CPE	2 x vCPU, 512 Mb RAM
vCPE-4	vKESR-M1 image	wan DHCP lan 10.20.4.1 overlay 172.16.1.4	CPE	2 x vCPU, 512 Mb RAM
vCPE-51	vKESR-M1 image	wan DHCP lan 10.20.5.2 / vIP 10.20.5.1 overlay 172.16.1.51	CPE	2 x vCPU, 512 Mb RAM
vCPE-52	vKESR-M1 image	wan DHCP lan 10.20.5.3 / vIP 10.20.5.1 overlay 172.16.1.52	CPE	2 x vCPU, 512 Mb RAM
R11	CentOS 7	wan 10.50.1.11 lan 10.1.4.1	DC border router	2 x vCPU, 2 GB RAM
R12	CentOS 7	wan 10.50.2.12 lan 10.1.5.1	DC border router	2 x vCPU, 2 GB RAM

Host	Operation system	IP address	Description	System requirements
R13	CentOS 7	dc-perim 10.1.3.13 oob 10.0.1.13 dc-lan1 10.1.1.1	DC core router	2 x vCPU, 2 GB RAM
R14	CentOS 7	wan 10.50.1.14 lan 10.1.3.1	DC border router and NAT	2 x vCPU, 2 GB RAM
ISP	CentOS 7	isp1 10.50.1.1 isp2 10.50.2.1 isp5 10.50.5.1 isp6 10.50.6.1 isp7 10.50.7.1 isp8 10.50.8.1	Emulation of ISP1-ISP8	2 x vCPU, 2 GB RAM
srv1	CentOS 7	10.1.1.11	WWW / DC server	2 x vCPU, 4 GB RAM
wst3	CentOS 7	DHCP 10.20.3.0/24	Site3 workstation	2 x vCPU, 4 GB RAM
wst4	CentOS 7	DHCP 10.20.4.0/24	Site4 workstation	2 x vCPU, 4 GB RAM
wst5	CentOS 7	DHCP 10.20.5.0/24	Site5 workstation	2 x vCPU, 4 GB RAM
mgmt	Windows Server 2022	10.0.1.22 10.1.1.22 10.1.3.22 10.50.1.22 10.20.3.22 10.20.4.22 10.20.5.22	Management workstation	6 x vCPU, 6 GB RAM



## 2.3. Network ports used by core system components

Table 2 shows the network ports used by SD-WAN gateways and CPE devices to communicate with the core components of the solution and to access the orchestrator web interface for administration.

Table 2 – Network ports used for communication of SD-WAN gateways and CPE devices with the SD-WAN management system.

Component	Ports	Description
SD-WAN orchestrator	TCP 443 / TLS	Access to the orchestrator web interface. CPE connection to the orchestrator.
SD-WAN controllers	TCP 6653-6656 / TLS	SD-WAN gateways and CPE devices connection to the controller over TLS. CPE device is connected by each WAN interface to a separate port of the controller: <ul style="list-style-type: none"> <li>• sdwan0 - 6653</li> <li>• sdwan1 - 6654</li> <li>• sdwan2 - 6655</li> <li>• sdwan3 - 6656</li> </ul>
Zabbix	TCP 85 / TLS TCP 10051 / TLS	Access to the Zabbix web interface. CPE Zabbix agent connections to the monitoring system.
SD-WAN gateways	UDP 4800-4803	Data traffic

## 2.4. SD-WAN containers' external connections diagram

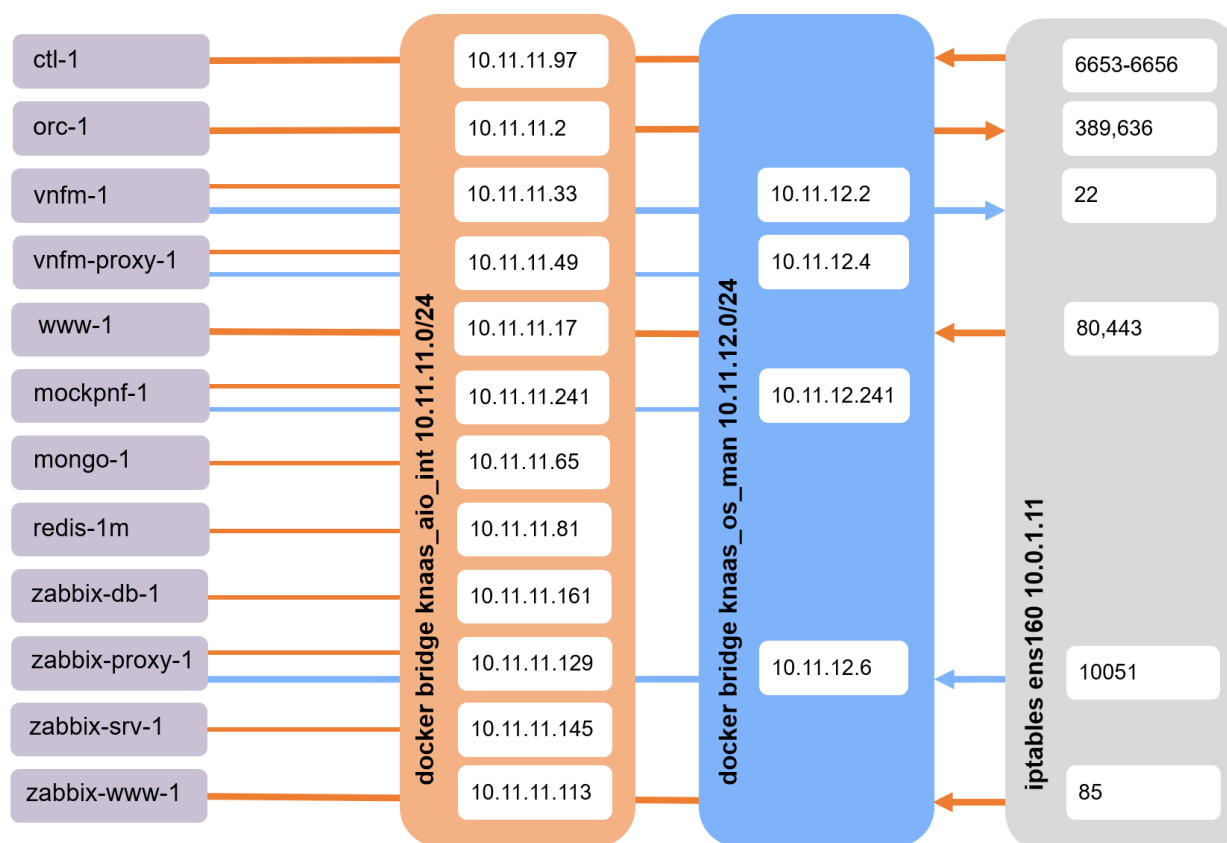


Figure 3 - SD-WAN containers' external connections.

SD-WAN containers will be deployed on the orc-1 host. The deployment playbooks create two Docker networks for the containers: **knaas\_aio\_int** (10.11.11.0/24) and **knaas\_os\_man** (10.11.12.0/24).

**knaas\_aio\_int** is the primary network and is used for communication between containers, as well as for communication with external hosts. **knaas\_os\_man** is intended for communication between the central components of the solution and CPEs. This network is used for CPE management and monitoring.

In addition, deployment playbooks create iptables rules. Iptables rules are added to the DOCKER\_USER chain to allow the following TCP connections:

- Inbound connections to the ctl-1 container on ports 6653-6656 (TLS connections from the CPE devices to the controller)
- Outbound connections from the orc-1 container on ports 389,636 (LDAP/LDAPS connections to the LDAP server)
- Outbound connections from the vnfm-1 container on port 22 (SSH console to CPE from the SD-WAN orchestrator interface)
- Inbound connections to the www-1 container on ports 80 and 443 (HTTPS/TLS connection to the orchestrator web interface and connections from CPEs to orchestrator)
- Inbound connections to the zabbix-proxy-1 container on port 10051 (CPE and VNF monitoring)
- Inbound connections to the zabbix-www-1 container on port 85 (HTTPS/TLS connection to the Zabbix monitoring system web interface)

## 2.5. Software versions

Table 3 - Versions of Kaspersky SD-WAN software used in this PoC

SD-WAN component	Version
www	knaas-www:2.25.03.release.57.gbl.amd64_en-US_ru-RU
orc	knaas-orc:2.25.03.release.39.gbl.amd64_en-US_ru-RU
mongo	mongo:5.0.7-r0amd64
ctl	knaas-ctl:2.25.03.release.17.gbl.amd64_en-US_ru-RU
vnfm	knaas-vnfm:2.25.03.release.10.gbl.amd64_en-US_ru-RU
vnfm-proxy	knaas-vnfm-proxy:2.25.03.release.6.gbl.amd64_en-US_ru-RU
redis	redis:6.2.7-r0.amd64
zabbix-www	zabbix-web-nginx-mysql:6.0.32-r0.amd64
zabbix-proxy	zabbix-proxy:6.0.32-r0.amd64
zabbix-srv	zabbix-server:6.0.32-r0.amd64
zabbix-db	mariadb-ha:11.1.6.amd64
syslog	syslog-ng:3.30.1-r1.amd64
vCPE	knaas-cpe_2.25.03.release.28
mockpnf	mockpnf: 2.23.09.amd64
orc1 host	Ubuntu 22.04.05 LTS Server
installer	knaas-installer_2.25.03.release.10.gbl.amd64_en-US_ru-RU

## 2.6. Hardware requirements for the Kaspersky SD-WAN solution

Table 4 - Hardware requirements for management of up to 50 CPE devices.

Host	CPU (hyper-threading), cores	RAM, GB	Disk, GB, SSD
orc1	16 cores / 16 vCPU (HT disabled) / 32 vCPU (HT enabled)	32	50 used in PoC / 256 Recommended

For more information, please refer to Kaspersky SD-WAN Online Help:  
<https://support.kaspersky.com/help/SD-WAN/2.4/en-US/239105.htm>

### 3. Kaspersky SD-WAN solution components: installation and configuration

To deploy the SD-WAN solution, you need to create a virtual machine (in this guide, the hostname is set to `orc1`) and install the Ubuntu 22.04.05 operating system. If the virtual machine is already created, go to section 3.2.

The Ubuntu 22.04.05 LTS Server Linux is used for the installation:

<https://releases.ubuntu.com/jammy/ubuntu-22.04.5-live-server-amd64.iso>

#### 3.1. Operation system installation on `orc1` host

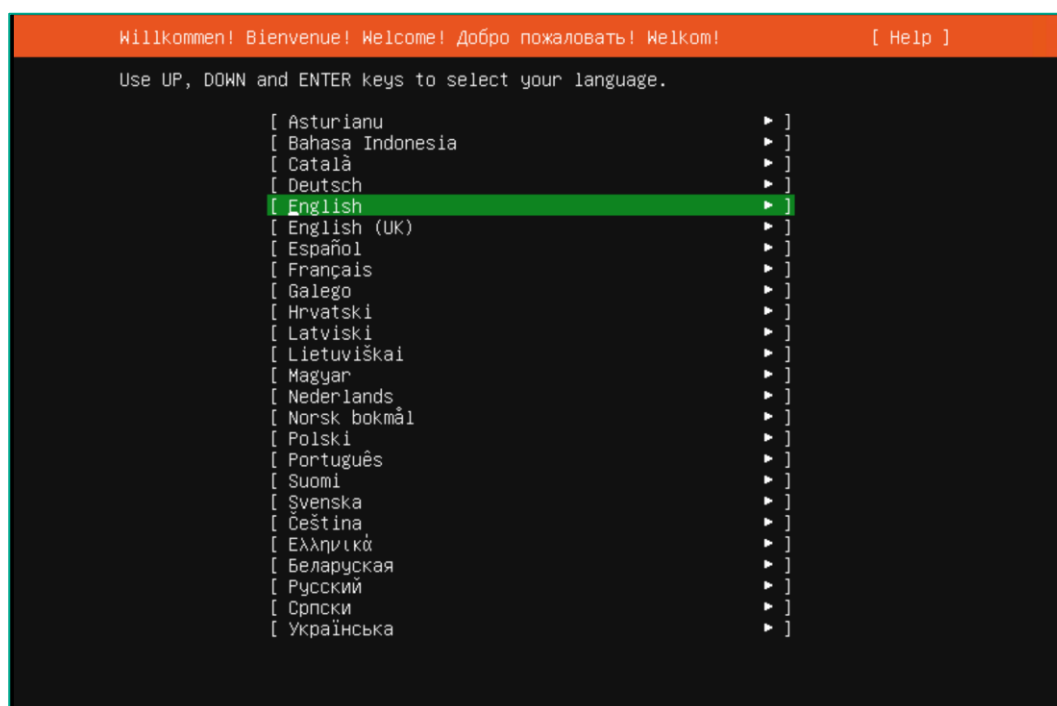
3.1.1. Create the virtual machine **orc1**.

Set the CPU, RAM, and disk resources according to Table 4.

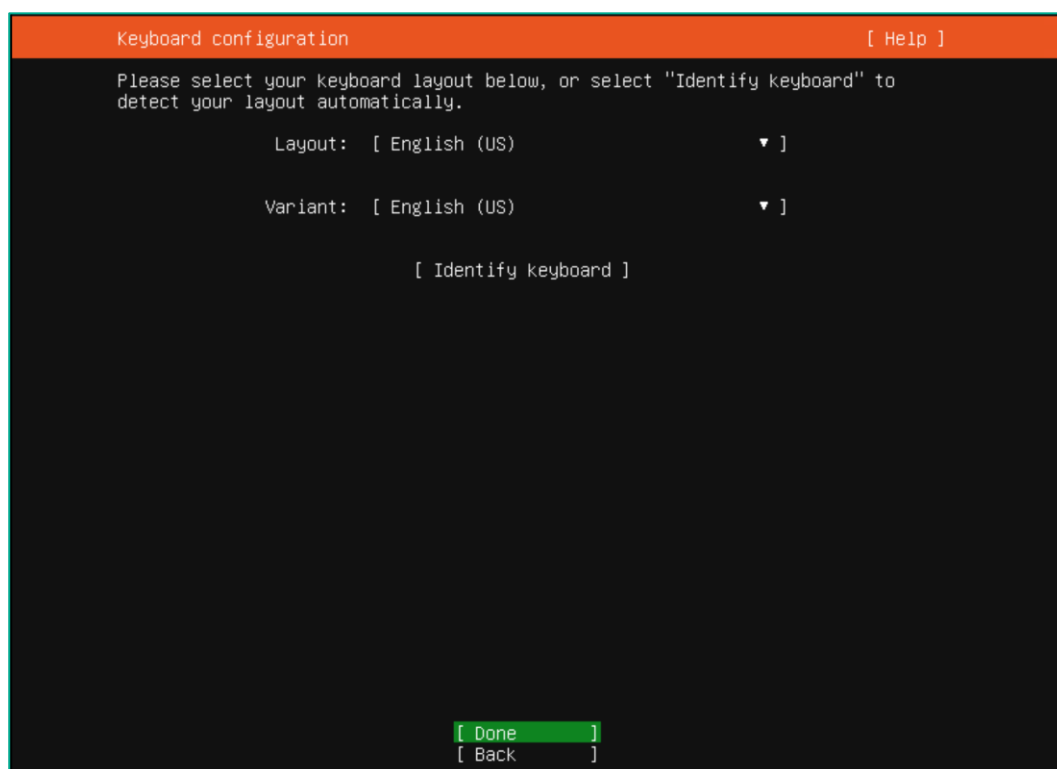
3.1.2. Start the operating system installation on host **orc1**.

Use **Ubuntu 22.04.05 LTS Server** image for installation.

Select language: **English** (default value).

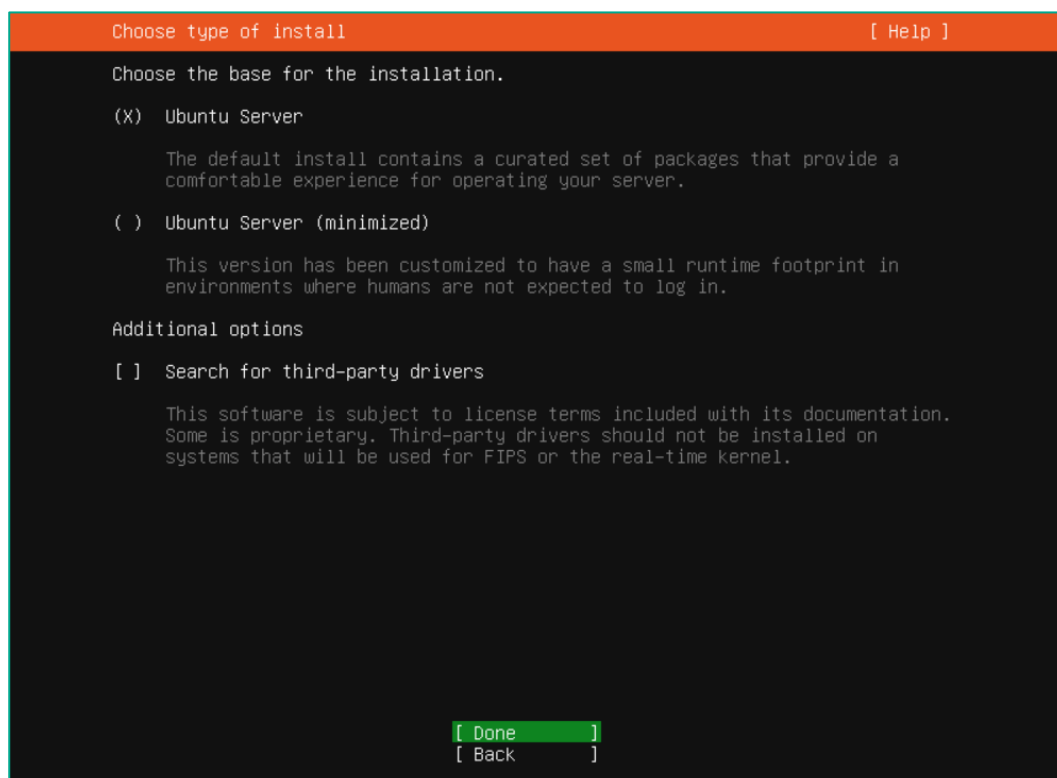


3.1.3. Select the keyboard layout: **US/US** (default).



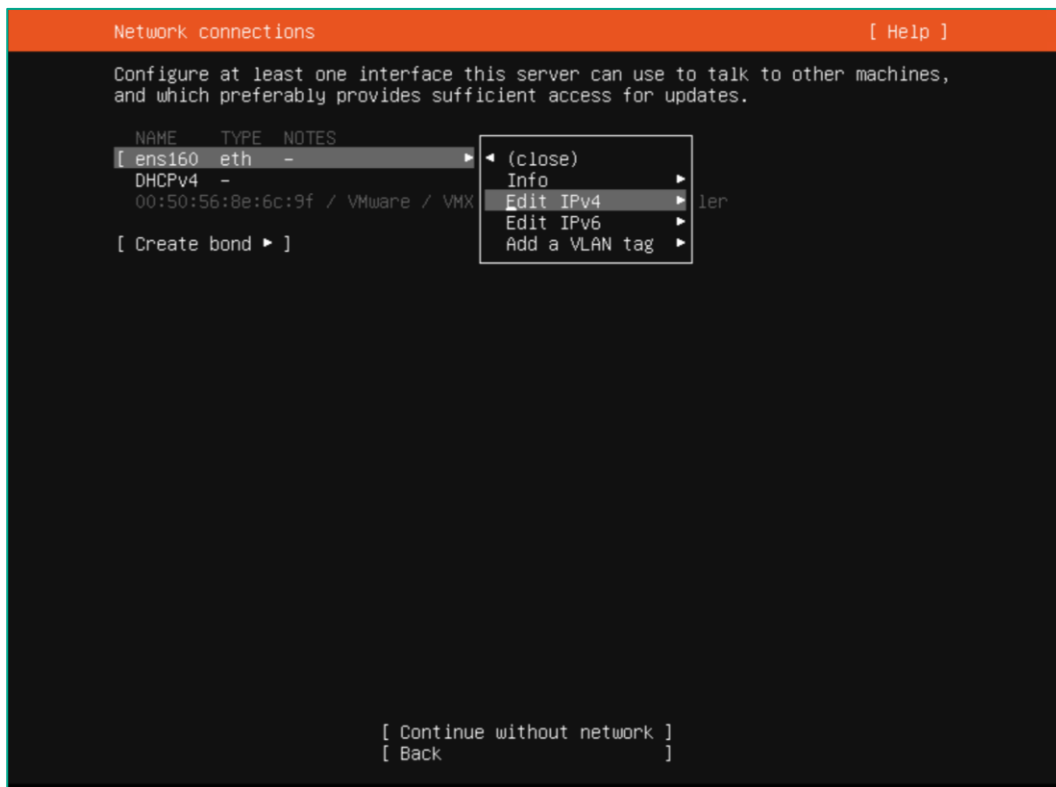
3.1.4. Select base for the installation – **Ubuntu Server**.

Select **Done**.

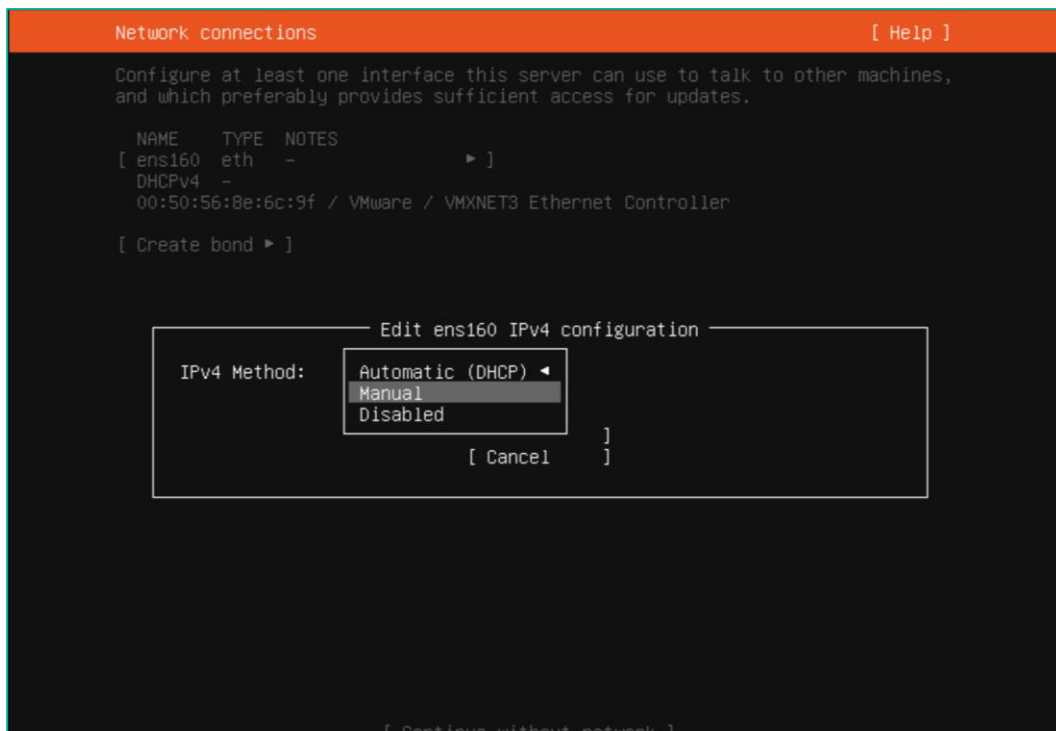


### 3.1.5. Configure the network interface on host orcl.

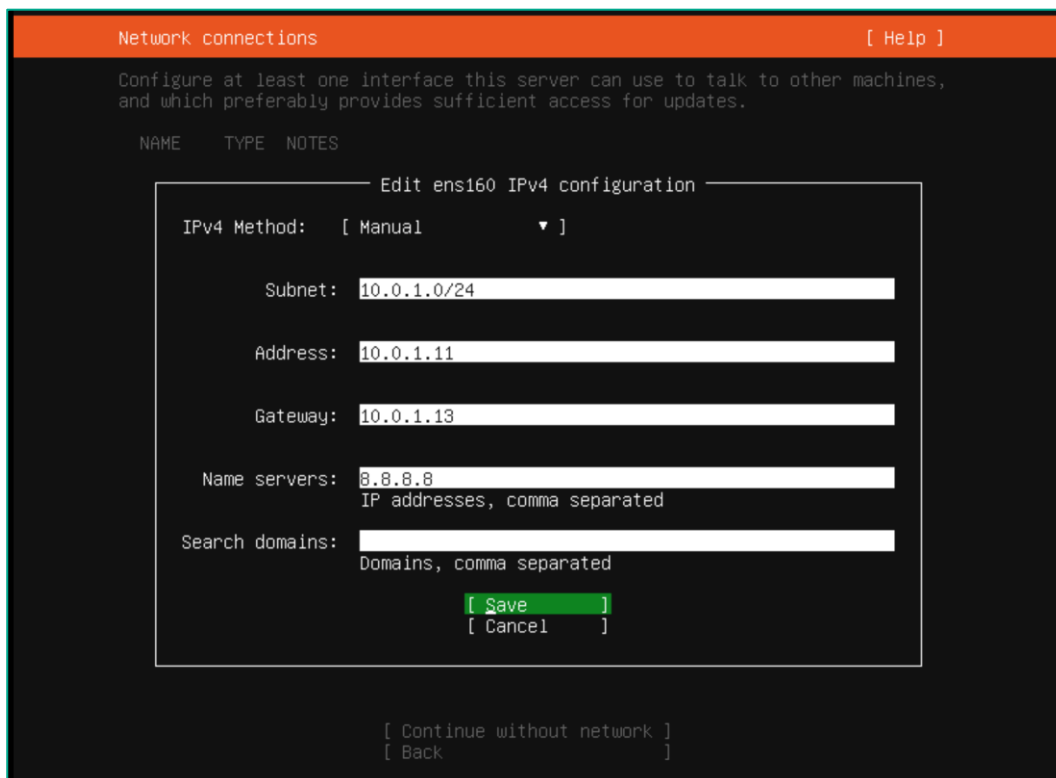
**Edit** the network interface **IPv4** settings (**ens160** in the example).



Select **Manual** IPv4 configuration.

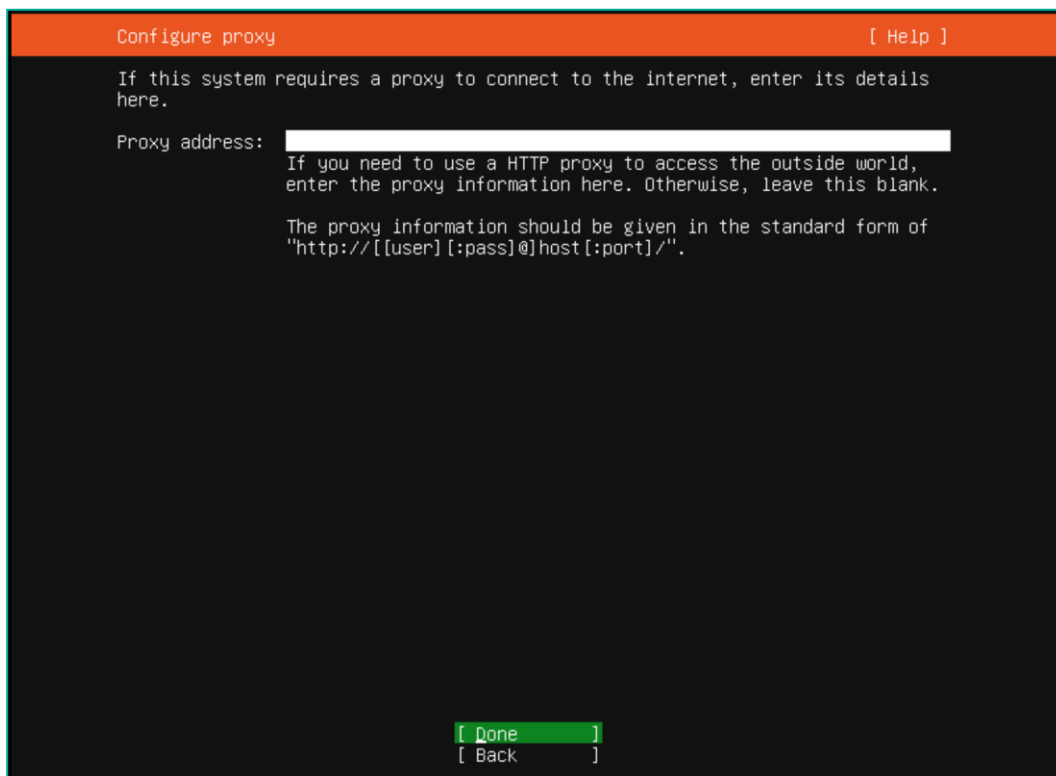


Configure the IPv4 parameters of the network interface according to the IP address plan and select **Save**.



3.1.6. Skip (configure if necessary) proxy configuration.

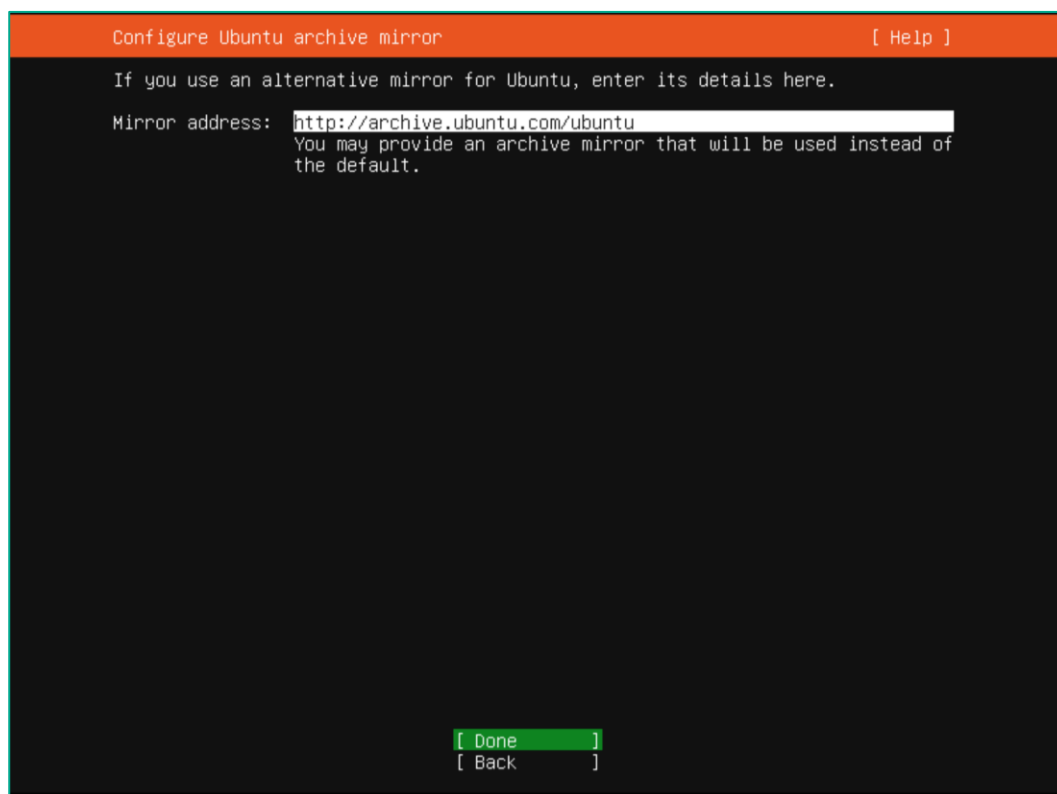
Select **Done**.





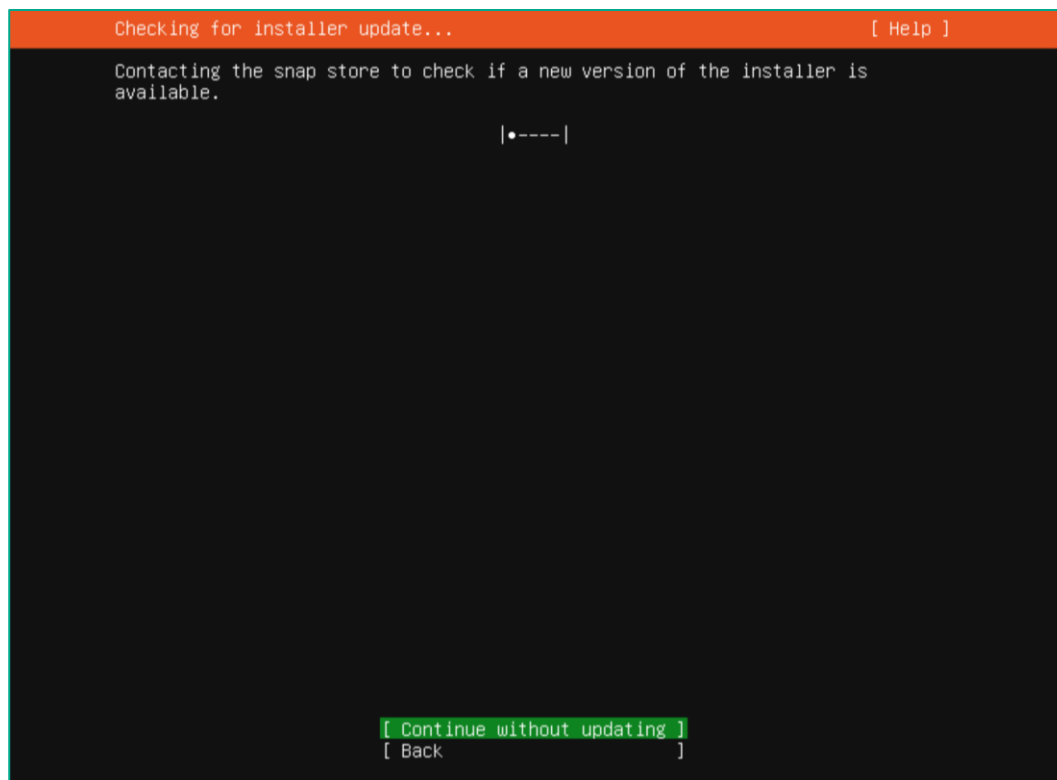
### 3.1.7. Configure the Ubuntu archive mirror.

Use the default value, select **Done**.

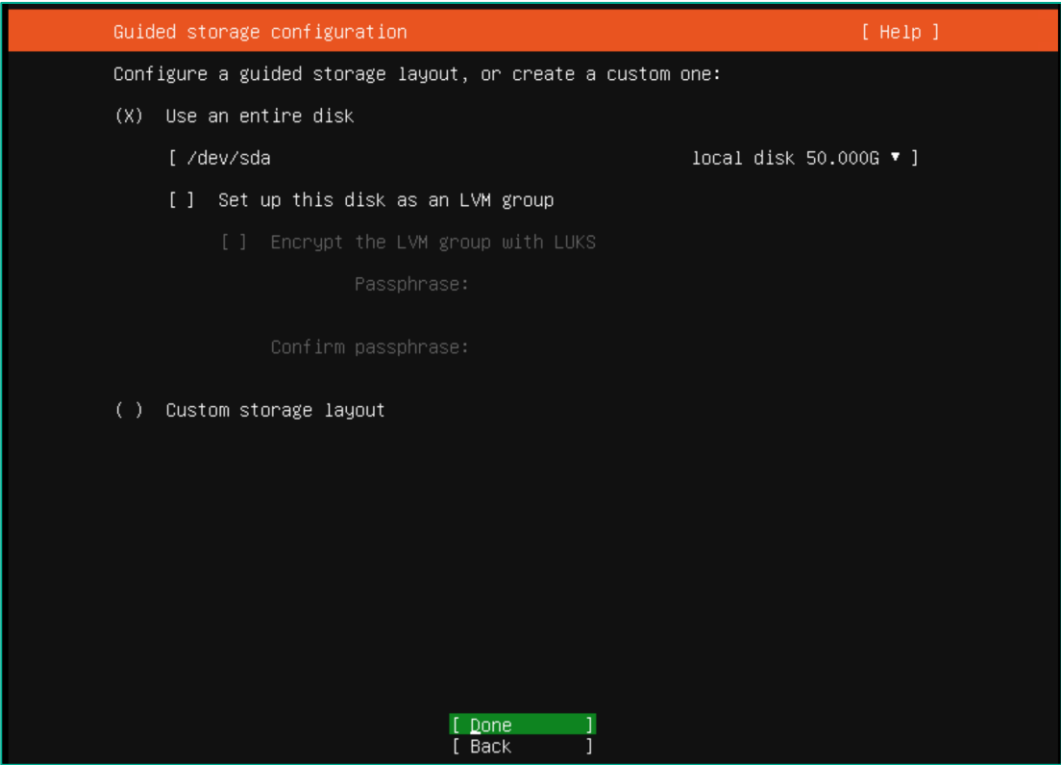


### 3.1.8. Skip installer update.

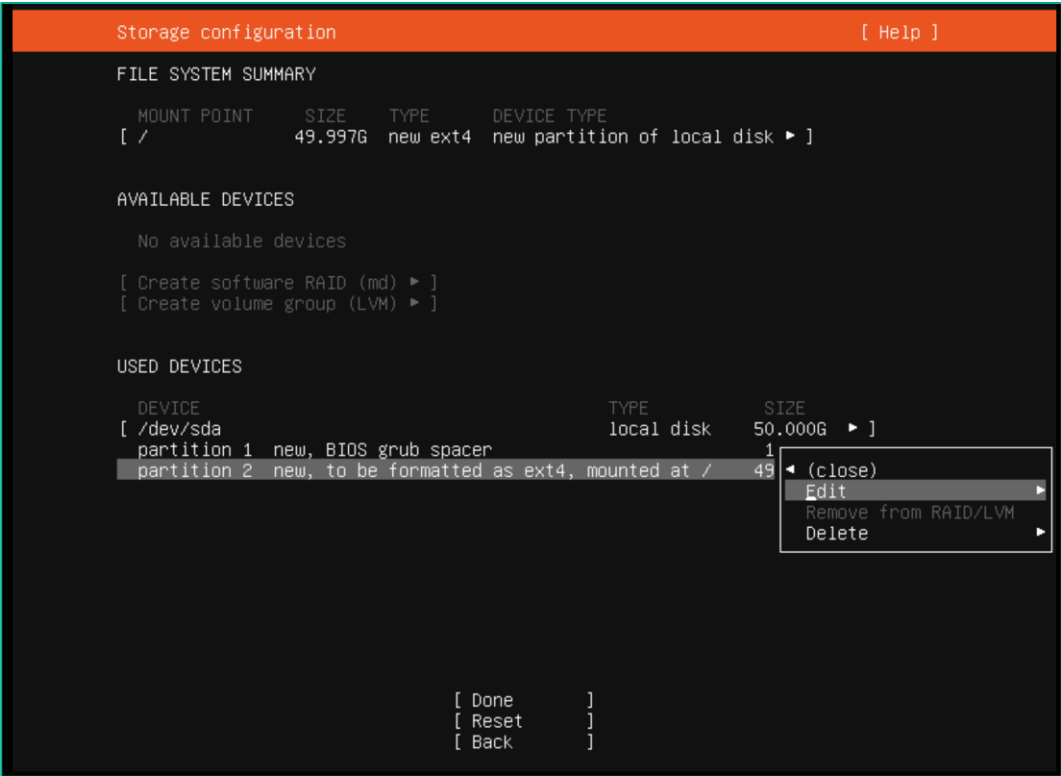
Select **Continue without updating**.



3.1.9. Select to **Use the entire disk** for the installation, without creating logical volume groups.  
Select **Done**.



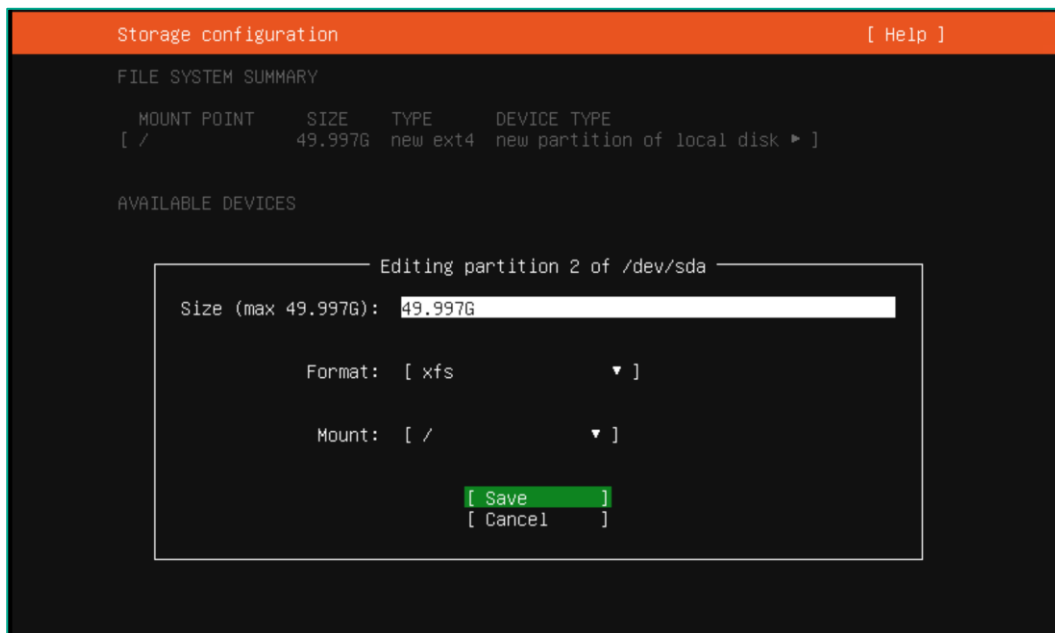
3.1.10. Configure storage.  
Edit the root partition (select **partition 2** and **Edit**).



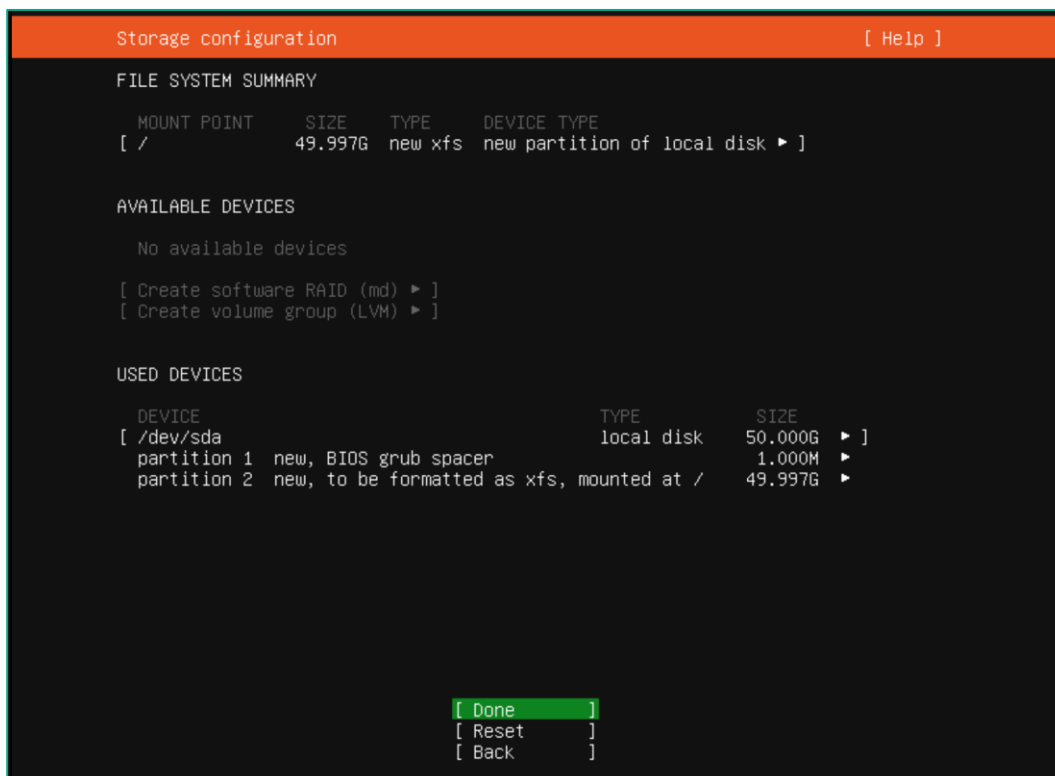
Set the file system format to **XFS**.

Select **Save**.

A **50GB** disk is used in this PoC. For 50xCPE devices, it's recommended to use 256GB.

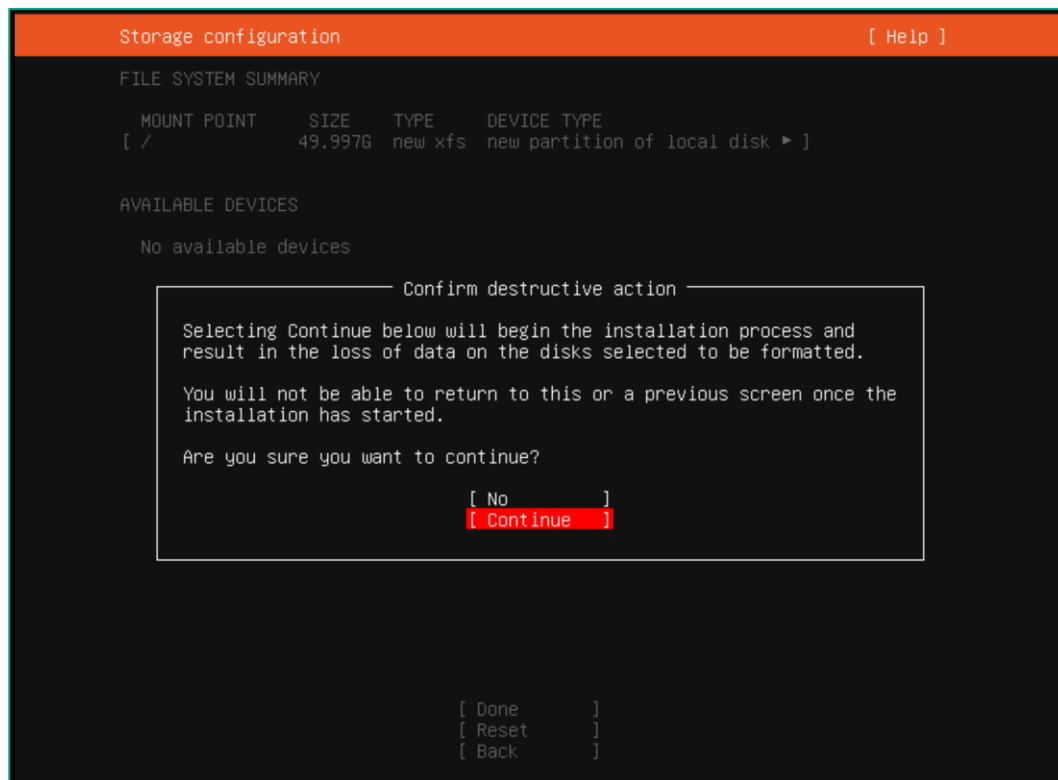


Confirm the settings, select **Done**.



3.1.11. Confirm the start of the installation process.

Select **Continue**.



### 3.1.12. Create a service user with the **sdwan username**.

This user is used when deploying the Kaspersky SD-WAN management system.

Set the **server name: orc1**

Profile setup [ Help ]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a password is still needed for sudo.

Your name: Kaspersky SD-WAN

Your server's name: orc1  
The name it uses when it talks to other computers.

Pick a username: sdwan

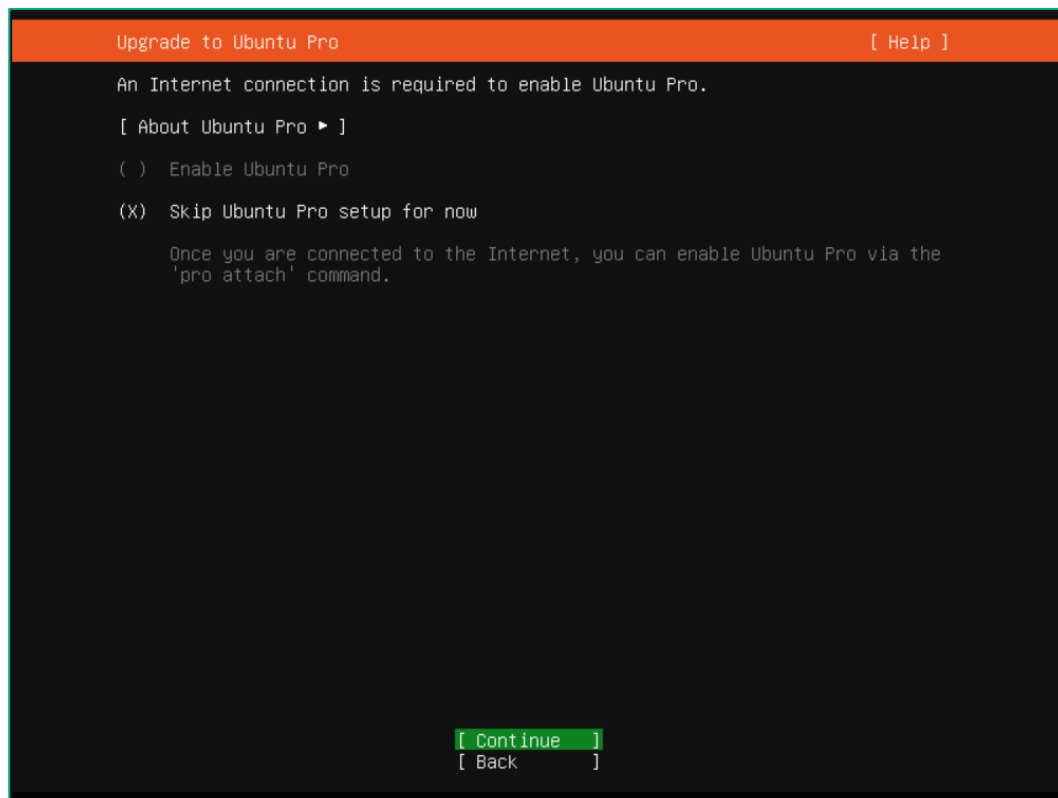
Choose a password: \*\*\*\*\*

Confirm your password: \*\*\*\*\*

[ Done ]

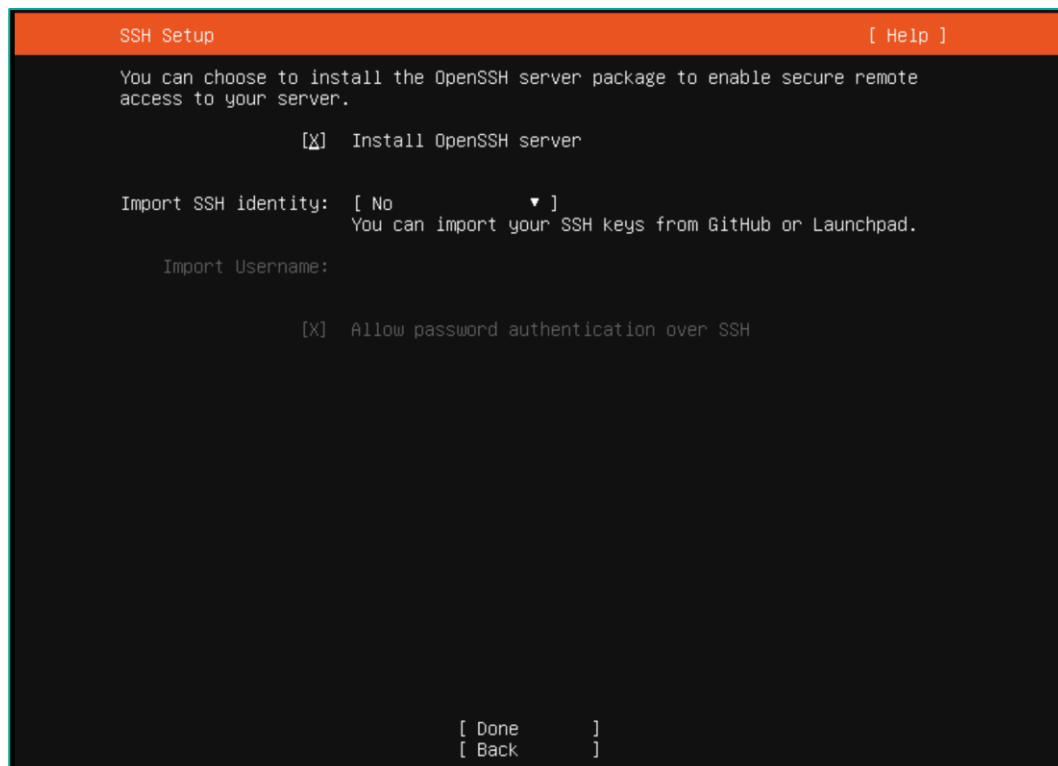
### 3.1.13. Skip Ubuntu Pro.

Select **Continue**.



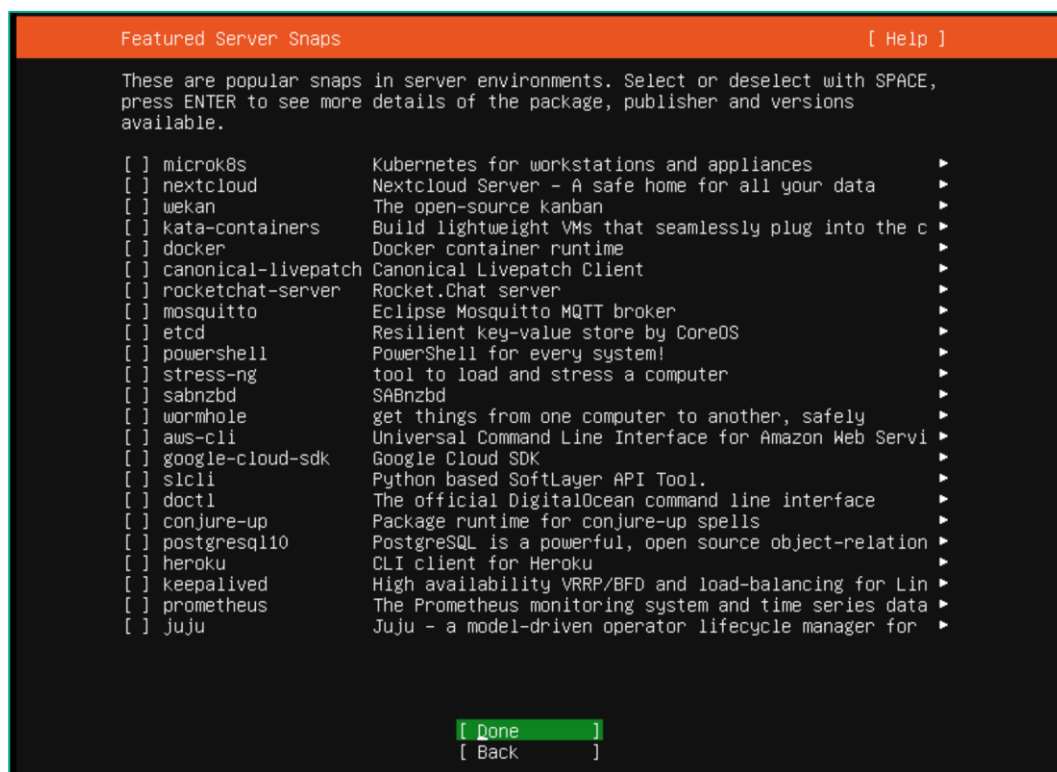
### 3.1.14. Add the OpenSSH service installation.

Check **Install OpenSSH server**, select **Done**.

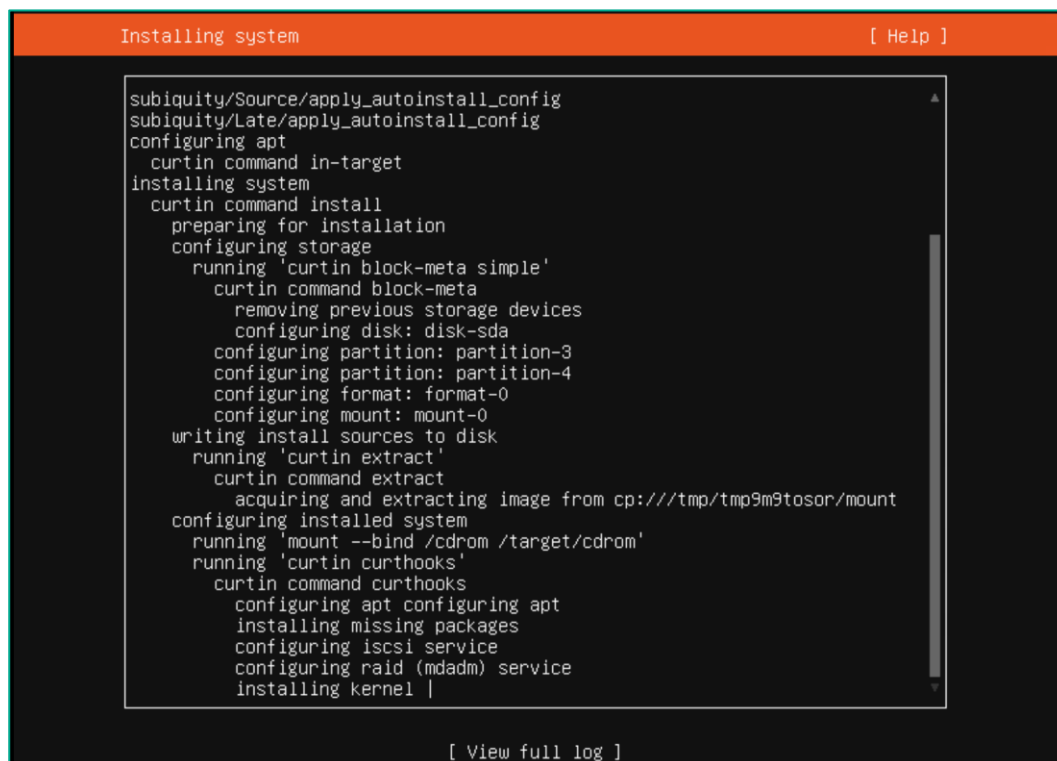


## 3.1.15. Skip installation of additional packages.

Select **Done** (the necessary packages will be installed later).



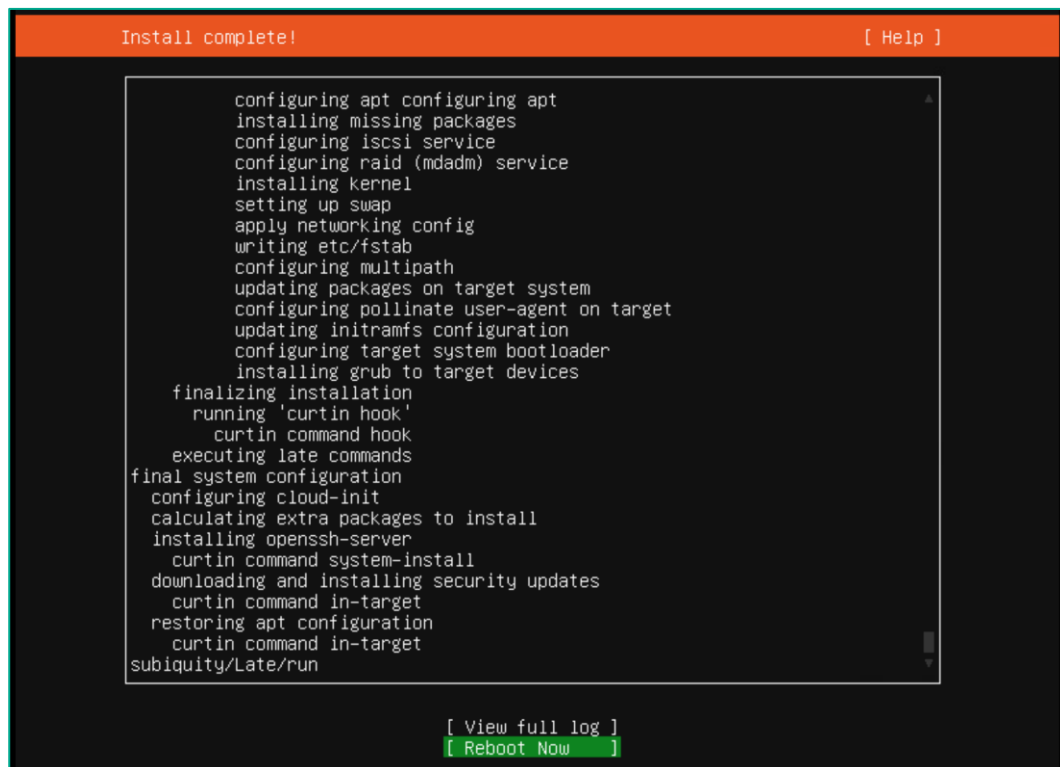
## 3.1.16. Wait for the system installation to start (a confirmation messages will appear).





## 3.1.17. Reboot orcl host.

Select **Reboot Now** to complete the installation.



## 3.2. Installation of Kaspersky SD-WAN Management system components

3.2.1. Check NTP status on the orc1 host.

Connect to the **orc1** host.

Check NTP status:

```
timedatectl status
```

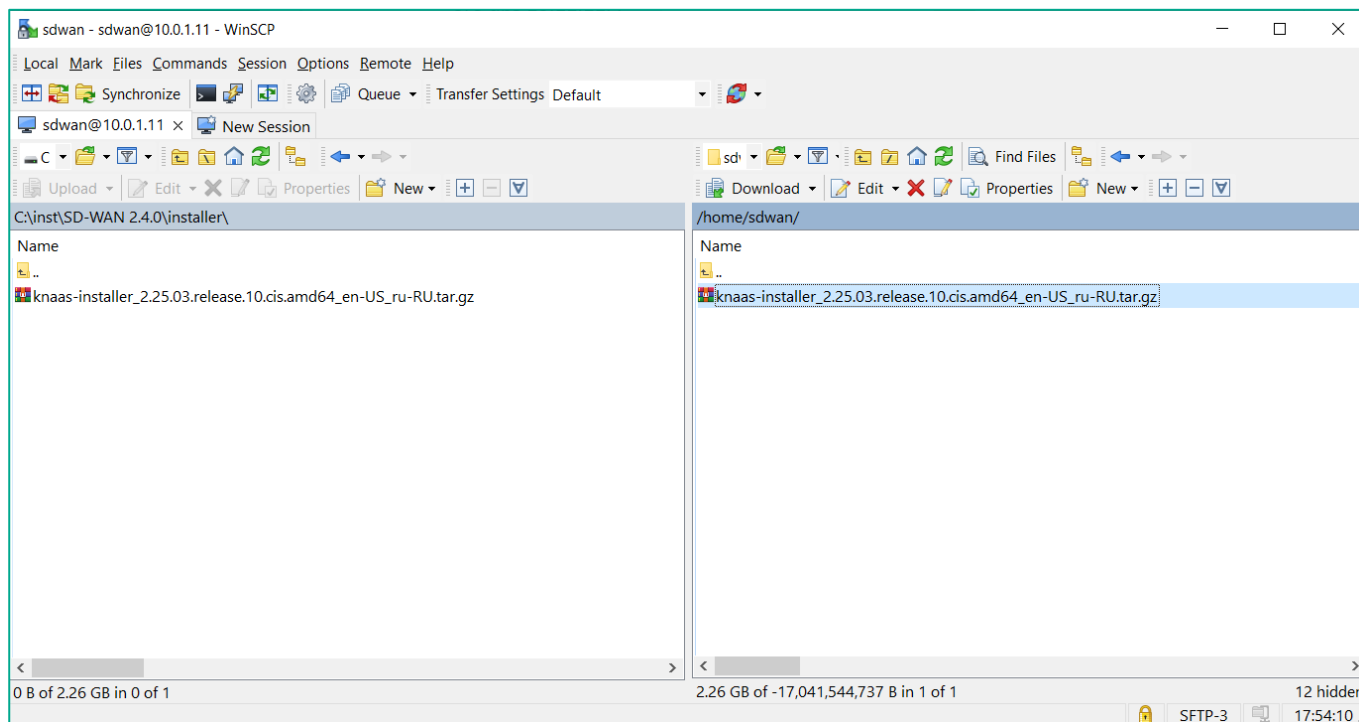
Clock must be synced:

System clock synchronized: **yes**

3.2.2. Upload the installation archive to the orc1 host1.

Upload the **knaas-installer.<release\_name>.gbl.amd64\_en-US\_ru-RU.tar.gz** archive to the home directory of the user **sdwan** on the host **orc1**. It includes Ansible playbooks for installing Kaspersky SD-WAN components.

**Note:** The user **sdwan** created in step 3.1.12 is used for installation. If a different user is used, the appropriate directory must be selected.



3.2.3. Unzip the installation archive to **sdwan** user home directory:

```
tar -xzf knaas-installer.<release_name>.gbl.amd64_en-US_ru-RU.tar.gz
```

Move to knaas-installer directory:

```
cd knaas-installer.<release_name>.gbl.amd64_en-US_ru-RU/
```

```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU
devices/baremetal/firmwares/KESR-M3/
devices/baremetal/firmwares/KESR-M3/knaas-cpe_2.25.03.release.91.efi.amd64-kesr-m3-k-4g-4s_en-US_ru-RU.tar.gz
devices/baremetal/firmwares/KESR-M4/
devices/baremetal/firmwares/KESR-M4/knaas-cpe_2.25.03.release.91.efi.amd64-kesr-m4-k-4g-2x_en-US_ru-RU.tar.gz
devices/baremetal/firmwares/KESR-M4/knaas-cpe_2.25.03.release.91.efi.amd64-kesr-m4-k-8g-4x_en-US_ru-RU.tar.gz
devices/baremetal/firmwares/KESR-M5/
devices/baremetal/firmwares/KESR-M5/knaas-cpe_2.25.03.release.91.efi.amd64-kesr-m5-k-4g-8x_en-US_ru-RU.tar.gz
devices/baremetal/firmwares/KESR-M5/knaas-cpe_2.25.03.release.91.efi.amd64-kesr-m5-k-8g-4x_en-US_ru-RU.tar.gz
devices/baremetal/firmwares/dpdk/
devices/baremetal/firmwares/dpdk/KESR-M5/
devices/baremetal/firmwares/dpdk/KESR-M5/knaas-cpe_2.25.03.release.91.efi.dpdk-kesr-m5-k-4g-8x_en-US_ru-RU.tar.gz
devices/
devices/firmwares/
devices/firmwares/vKESR-M1/
devices/firmwares/vKESR-M1/knaas-cpe_2.25.03.release.91.bios.amd64-vkesr-m1_en-US_ru-RU.tar.gz
devices/firmwares/vKESR-M2/
devices/firmwares/vKESR-M2/knaas-cpe_2.25.03.release.91.bios.amd64-vkesr-m2_en-US_ru-RU.tar.gz
devices/firmwares/vKESR-M3/
devices/firmwares/vKESR-M3/knaas-cpe_2.25.03.release.91.bios.amd64-vkesr-m3_en-US_ru-RU.tar.gz
devices/firmwares/vKESR-M4/
devices/firmwares/vKESR-M4/knaas-cpe_2.25.03.release.91.bios.amd64-vkesr-m4_en-US_ru-RU.tar.gz
devices/images/
devices/images/vKESR-M1/
devices/images/vKESR-M1/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m1.vKESR-M1-esxi.tar.gz
devices/images/vKESR-M1/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m1.vKESR-M1-kvm.tar.gz
devices/images/vKESR-M2/
devices/images/vKESR-M2/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m2.vKESR-M2-esxi.tar.gz
devices/images/vKESR-M2/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m2.vKESR-M2-kvm.tar.gz
devices/images/vKESR-M3/
devices/images/vKESR-M3/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m3.vKESR-M3-esxi.tar.gz
devices/images/vKESR-M3/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m3.vKESR-M3-kvm.tar.gz
devices/images/vKESR-M4/
devices/images/vKESR-M4/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m4.vKESR-M4-esxi.tar.gz
devices/images/vKESR-M4/knaas-cpe_2.25.03.release.91.combined.amd64-vkesr-m4.vKESR-M4-kvm.tar.gz
sdwan@orc1:~$ cd knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU/
sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU$
```

### 3.2.4. Update package lists and upgrade packages.

```
sudo apt update && sudo apt upgrade --yes
```

### 3.2.5. Install the required packages before running the installation playbooks.

Install PIP:

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

Install the required packages using PIP (Ansible, PyMongo, Docker):

```
pip3 install -U --user -r requirements.txt
```

Add **\$HOME/.local/bin** to the **PATH** variable (required for Ansible to work properly):

```
echo 'export PATH=$PATH:$HOME/.local/bin' >> ~/.bashrc
```

Execute **.bashrc**, to apply the **PATH** variable:

```
source ~/.bashrc
```

Verify that Ansible is running properly:

```
ansible --version
```

### 3.2.6. Configure the installation parameters of the Kaspersky SD-WAN management system.

Copy the base inventory file with variables (in this guide the **poc\_aio.yml** file will be used):

```
cp inventory/external/pnf/local.yml /home/sdwan/poc_aio.yml
```

Open the **poc\_aio.yml** configuration file for editing:

```
vi /home/sdwan/poc_aio.yml
```

Press **i** for editing, after making changes, press **esc** and enter **:wq** to save the changes.

Set the following installation parameters:

- Add the internal and public IP addresses of host orc1 (**10.0.1.11** and **10.50.1.14**) to the **san\_list: ip** section. These addresses will be added to the Subject Alternative Name (SAN) of the SD-WAN orchestrator certificate. Keep the address **10.11.12.1** in this section, it will be used when connecting the controller to the orchestrator.
- Add the domain name of host orc1 (sdwan.local as an example) to the **san\_list: dns** section. The domain names will be added to the SAN of the SD-WAN orchestrator certificate.
- Path to save database and vault passwords: **vault\_passwords\_dirname:**  
**/home/sdwan/passwords/.**
- Path to save certificates: **ssl: path\_local:** **/home/sdwan/ssl.**

```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU
---
# Local All-In-One PNF Config Example
version: "2.25.03.0"

nodes:
  node_1:
    ip: 127.0.0.1
    knaas_aio_int:
      base: 10.11.11
      mode: bridge
    knaas_os_man:
      base: 10.11.12
      mode: bridge

external:
  vault_passwords_dirname: "/home/sdwan/passwords" # The directory where the keystore.yml and vault_password.txt f
  files are stored.
  ssl:
    san_list:
      ip:
        - "10.11.12.1"
        - "10.0.1.11"
        - "10.50.1.14"
      dns:
        - sdwan.local
    path_local: "/home/sdwan/ssl"

docker:
  local_path_to_images: "../images" # Directory where ansible will search docker images
  remote_path_to_images: "/tmp" # Directory where ansible will store files on remote VMs

syslog:
  docker_memory_limit: 1024 # in MegaBytes, depends on CPE count, <=250-1024,<=2K-4096M,<=5K-6144,<=10K-8192
  max_log_size: 32 # in GigaBytes, depends on customer requirements
  state: enabled
```

3.2.7. Prepare the orc1 host for installation by running the bootstrap playbook, which will install the required packages.

Set the EULA acceptance parameter:

```
export KNAAS_EULA_AGREED="true"
```

Run the installation playbook - you will be asked for a privilege escalation password:

```
ansible-playbook -i inventory/generic -e "@/home/sdwan/poc_aio.yml" -e "@inventory/external/images.yml"
-K knaas/utilities/node_prepare/bootstrap.yml
```

There must be no failed tasks during the installation.

```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU

TASK [KNAAS | Restart Docker service] *****
skipping: [127.0.0.1]

TASK [KNAAS | PREPARE | Restore daemon.json from backup] *****
skipping: [127.0.0.1]

TASK [KNAAS | Restart Docker service if daemon.json restore] *****
skipping: [127.0.0.1]

PLAY [KNAAS | Docker prepare] *****

TASK [KNAAS | Docker prepare | Start and enable Docker] *****
ok: [127.0.0.1]

TASK [KNAAS | Docker prepare | Add current user to the docker group] *****
ok: [127.0.0.1]

TASK [KNAAS | Docker prepare | Reset ssh connection to apply user changes] *****
[WARNING]: Reset is not implemented for this connection

PLAY [KNAAS | Node prepare] *****

TASK [KNAAS | Node prepare | Update pip3] *****
changed: [127.0.0.1]

TASK [KNAAS | Node prepare | Install python packages] *****
changed: [127.0.0.1]

PLAY RECAP *****
127.0.0.1      : ok=11  changed=5  unreachable=0  failed=0  skipped=18  rescued=0  ignored=0
localhost     : ok=4   changed=1  unreachable=0  failed=0  skipped=0   rescued=0  ignored=0

sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU$
```

3.2.8. Apply permissions for **sdwan** user.

The **sdwan** user must be added to the docker group in order to run containers. During the execution of the **bootstrap** playbook, the user was added to the group, but the user shell must be invoked again to apply the changes:

```
su sdwan
```

### 3.2.9. Perform a check before installing the Kaspersky SD-WAN solution.

Run the **pre-flight** playbook to perform pre-installation checks, you will be asked for a privilege escalation password. There must be no failed tasks during the checks:

**ansible-playbook -K knaas/utilities/pre-flight.yml**

```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU
msg: Success

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Docker Permission Check] *****
ok: [Toolserver]

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Docker Permission Check Results] *****
ok: [Toolserver] =>
  msg: Docker is accessible by current user

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Java Installed Check] *****
changed: [Toolserver]

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Java Installed Check Results] *****
ok: [Toolserver] => changed=false
  msg: Success

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Java Installed Check Version] *****
ok: [Toolserver] => changed=false
  msg: Success

PLAY [KNAAS | Utilities | Pre Flight Toolserver Checks] *****

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Make Installed Check] *****
changed: [localhost]

TASK [KNAAS | Utilities | Pre Flight Toolserver Checks | Make Installed Check Results] *****
ok: [localhost] => changed=false
  msg: Success

PLAY RECAP *****
Toolserver      : ok=8    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU$
```

## 3.2.10. Run the Kaspersky SD-WAN management system components installation playbook.

Start installing the Kaspersky SD-WAN management system. During installation, iptables firewall rules will be configured, certificates of the certification center and solution components will be generated, and Kaspersky SD-WAN management system containers will be launched.

Run the **knaas-install** installation playbook - you will be asked for privilege escalation password:

```
ansible-playbook -i inventory/generic -e "@/home/sdwan/poc_aio.yml" -e "@inventory/external/images.yml"
-K knaas/knaas-install.yml
```

Wait for the Kaspersky SD-WAN installation playbook to finish. There must be no failed tasks during the installation.

```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU

TASK [KNAAS | INSTALL | KNaas Controller | Deploy Controller] *****
changed: [ctl1-1]

TASK [KNAAS | INSTALL | KNaas Controller | Deploy Controller | PNF] *****
changed: [ctl1-1]

PLAY RECAP *****
127.0.0.1      : ok=17  changed=10  unreachable=0  failed=0  skipped=8  rescued=0  ignored=0
ctl1-1        : ok=33  changed=22  unreachable=0  failed=0  skipped=35  rescued=0  ignored=0
mongo-1       : ok=32  changed=15  unreachable=0  failed=0  skipped=33  rescued=0  ignored=2
orc-1         : ok=41  changed=27  unreachable=0  failed=0  skipped=32  rescued=0  ignored=0
redis-1m      : ok=18  changed=10  unreachable=0  failed=0  skipped=46  rescued=0  ignored=1
syslog-1      : ok=17  changed=11  unreachable=0  failed=0  skipped=38  rescued=0  ignored=0
vnfm-1        : ok=24  changed=18  unreachable=0  failed=0  skipped=34  rescued=0  ignored=0
vnfm-proxy-1  : ok=19  changed=12  unreachable=0  failed=0  skipped=36  rescued=0  ignored=0
www-1         : ok=20  changed=13  unreachable=0  failed=0  skipped=39  rescued=0  ignored=0
zabbix-db-1   : ok=24  changed=12  unreachable=0  failed=0  skipped=41  rescued=0  ignored=1
zabbix-proxy-1 : ok=24  changed=14  unreachable=0  failed=0  skipped=33  rescued=0  ignored=0
zabbix-srv-1  : ok=24  changed=15  unreachable=0  failed=0  skipped=32  rescued=0  ignored=0
zabbix-www-1  : ok=22  changed=13  unreachable=0  failed=0  skipped=33  rescued=0  ignored=0

sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU$
```

The installation process will generate passwords for databases and certificates. They will be saved in **/home/sdwan/passwords/keystore.yml** (path was set in step 3.2.6) and encrypted with ansible-vault. The password for vault will also be generated and saved in **/home/sdwan/passwords/vault\_password.txt**.

**Note:** Copy the certificates, keystore files, poc\_aio.yml and vault password for future use!

## 3.2.11. Clear bash history:

```
history -c && history -w
```



3.2.12. If the Kaspersky SD-WAN installation playbook needs to be run again, the installed containers must be removed.

To remove Kaspersky SD-WAN run the **knaas-teardown.yml** playbook:

```
ansible-playbook -i inventory/generic -e "@/home/sdwan/poc_aio.yml" -e "@inventory/external/images.yml"
-K knaas/knaas-teardown.yml
```

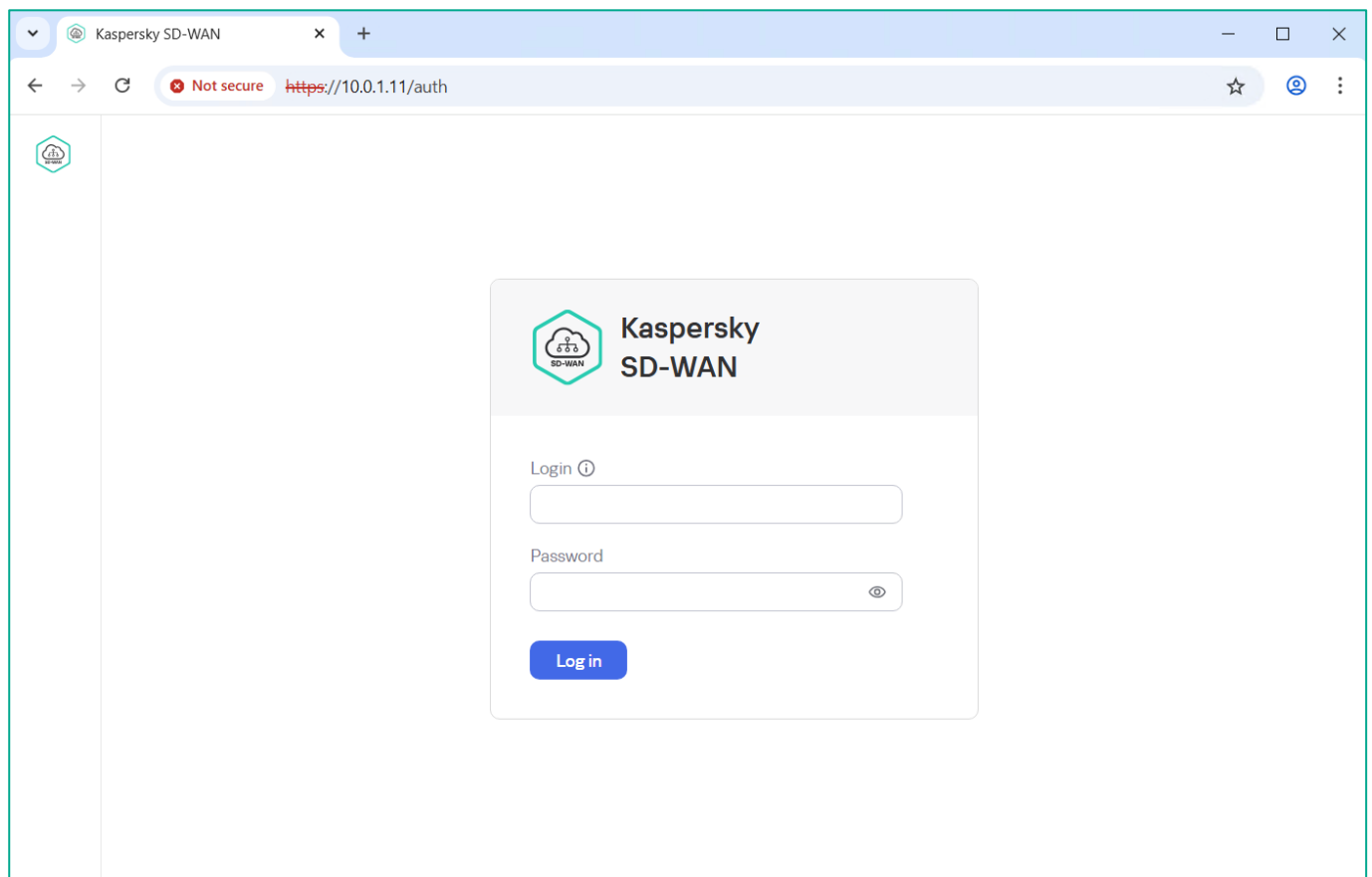
## 3.3. Connecting to the Kaspersky SD-WAN management console

### 3.3.1. Login to Kaspersky SD-WAN.

Login parameters:

- Orchestrator URL: <https://10.0.1.11>
- Default Login and Password: **admin** / **admin**.

**Note:** If you change the IP plan from 2.2, use the new IP address of the **orc1** host.



### 3.3.2. Change **Administrator** password.

Go to **Users**. Select **Administrator** user.

Click **Change password**.

The screenshot shows the Kaspersky SD-WAN management interface. On the left, a sidebar contains navigation icons. The main panel has tabs for 'Users', 'Permissions', 'Groups', 'LDAP connections', and 'Tools'. The 'Users' tab is active, displaying a table with columns 'Name', 'Tenant', and 'Role'. Two users are listed: 'Administrator Administrator' (Role: Administrator) and 'User User' (Role: Tenant). The 'Administrator Administrator' user is selected. A modal window titled 'User Online' is open, showing the 'Change password' tab. The modal contains fields for 'Login' (admin), 'Role' (Administrator), 'Two-factor authentication (Disabled)' (Off), 'Request confirmation is required' (Off), 'First name' (Administrator), 'Last name' (Administrator), and 'Email' (admin@example.com). At the bottom of the modal are 'Save' and 'Cancel' buttons.

Enter new password and click **Save**.

This screenshot shows the same interface as the previous one, but with the 'Password' modal open. The modal has fields for 'New password' and 'Password confirmation', both masked with dots. At the bottom of the modal are 'Save' and 'Cancel' buttons. The background shows the 'User' profile for 'Administrator' with the 'Change password' tab selected.

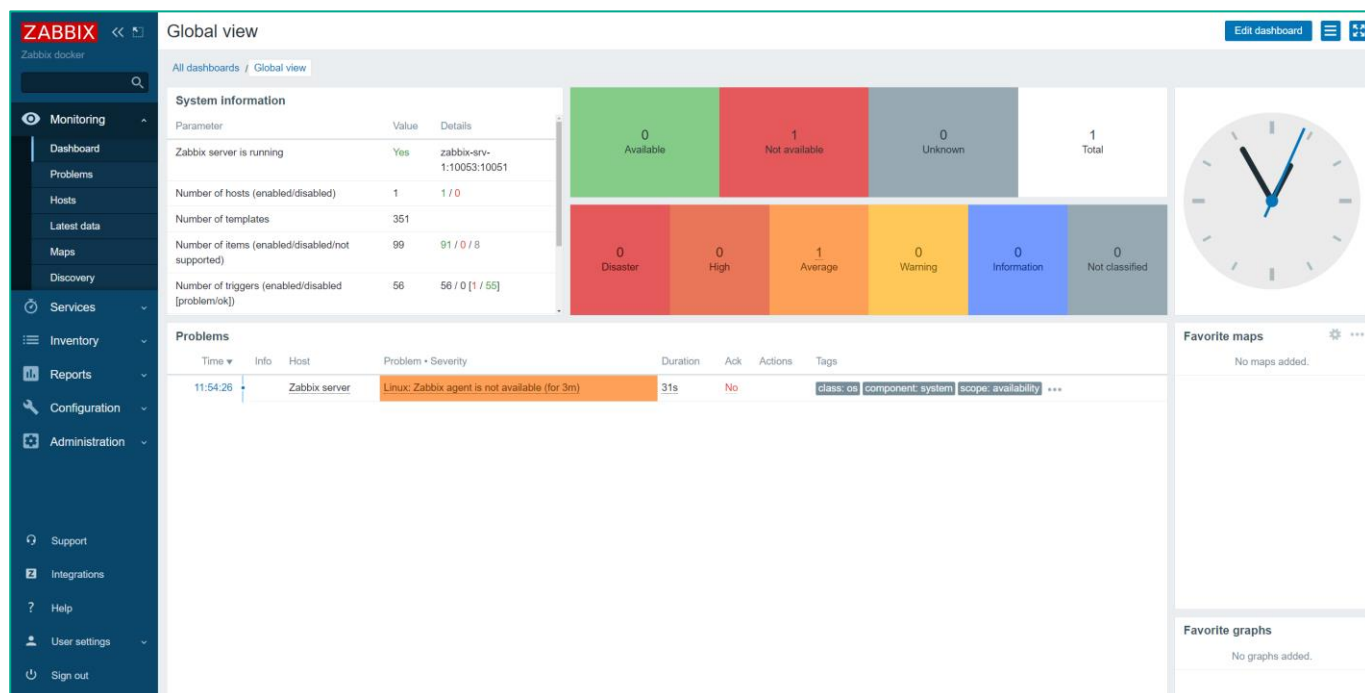
## 3.4. Logging in to the Zabbix monitoring system and set up configuration

### 3.4.1. Login to the Zabbix.

To connect to the Zabbix system, open the URL: <https://10.0.1.11:85>

Default login/password: **Admin** / **zabbix**.

**Note:** If you change the IP plan from 2.2, use the new IP address of the **orc1** host.



### 3.4.2. Change Administrator password.

Click **Administration** → **Users** → **Admin** → **Change password**.

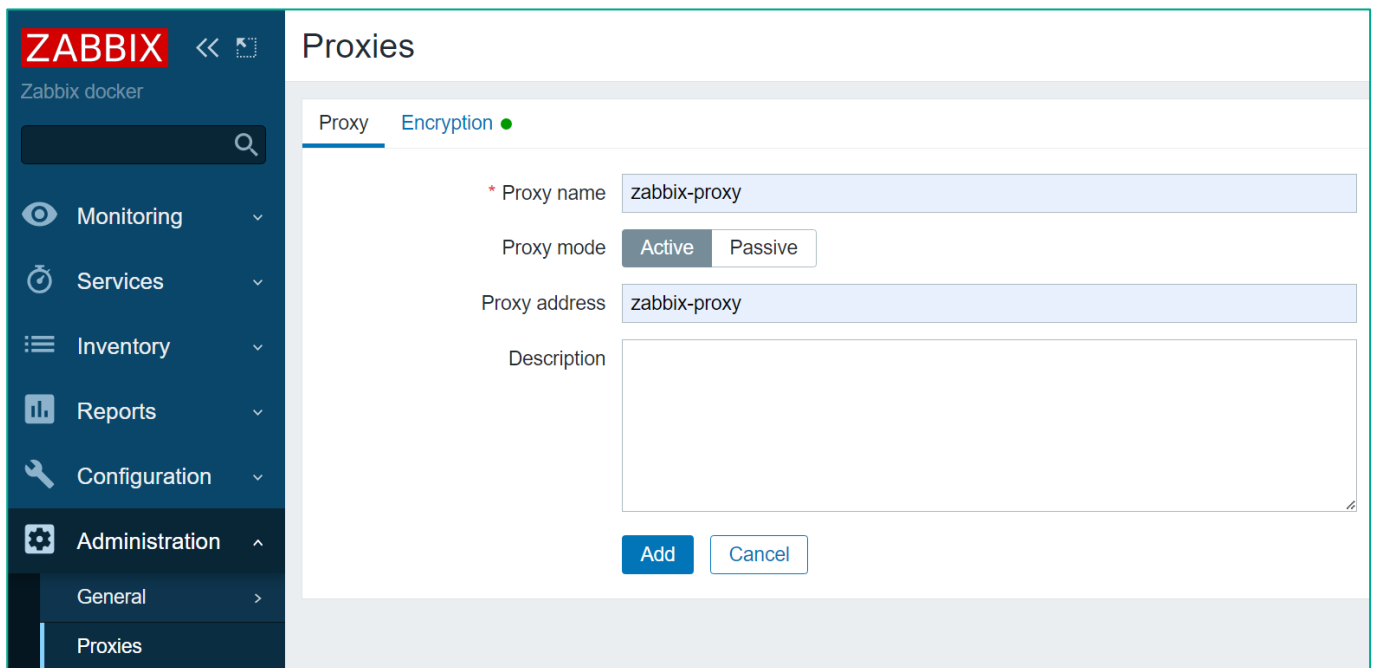
The screenshot shows the Zabbix web interface. On the left is a dark blue sidebar with the 'ZABBIX' logo and a search bar. Below the search bar are menu items: Monitoring, Inventory, Reports, Configuration, Administration (expanded), General, Proxies, Authentication, User groups, Users (selected), Media types, Scripts, Queue, and Support. The main content area is titled 'Users' and has three tabs: 'User' (active), 'Media', and 'Permissions'. The 'User' tab shows the configuration for the 'Admin' user. Fields include: Alias (Admin), Name (Zabbix), Surname (Administrator), Groups (Zabbix administrators), Password (Change password button), Language (English (en\_GB)), Theme (System default), Auto-login (checked), Auto-logout (unchecked, 15m), Refresh (30s), Rows per page (50), and URL (after login). At the bottom are 'Update', 'Delete', and 'Cancel' buttons.

After entering the new password, click **Update** to apply the settings.

### 3.4.3. Add Zabbix Proxy.

Go to **Administration** → **Proxies**, click **Create Proxy**.

Enter **zabbix-proxy** in both the **Proxy name** and **Proxy address** fields.



**ZABBIX** << Zabbix docker

Monitoring Services Inventory Reports Configuration Administration General Proxies

## Proxies

Proxy Encryption ●

\* Proxy name zabbix-proxy

Proxy mode Active Passive

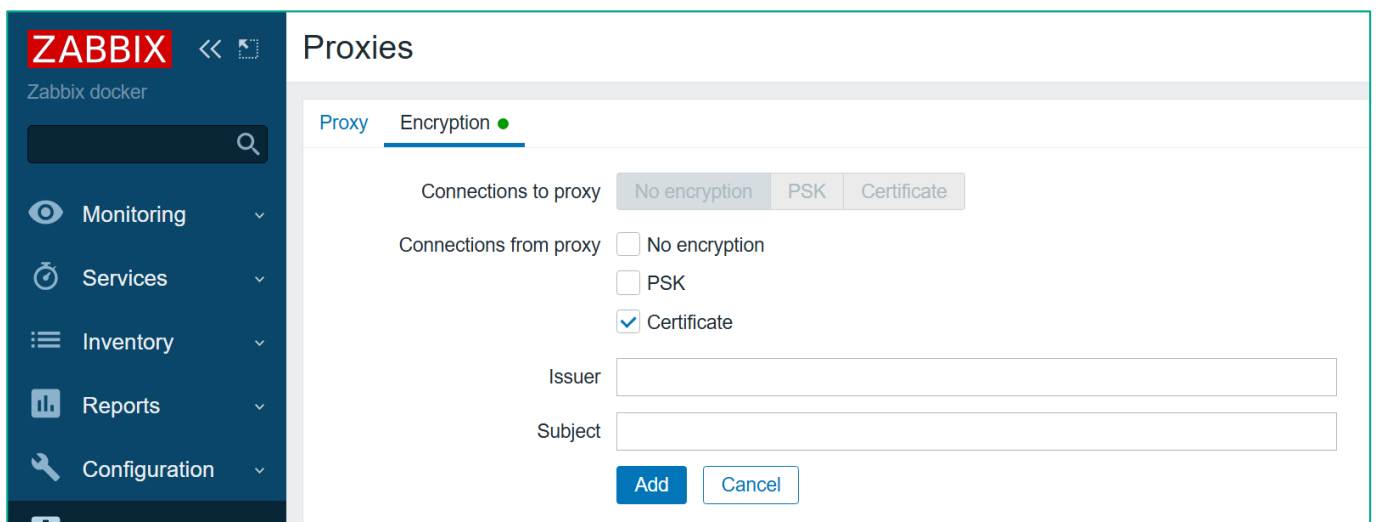
Proxy address zabbix-proxy

Description

Add Cancel

When connecting the proxy to the Zabbix server, encryption is used with the certificates created during the system installation.

Switch to the **Encryption** tab, check **Certificate**, and then click **Add**.



**ZABBIX** << Zabbix docker

Monitoring Services Inventory Reports Configuration Administration General Proxies

## Proxies

Proxy Encryption ●

Connections to proxy No encryption PSK Certificate

Connections from proxy ☐ No encryption ☐ PSK ☒ Certificate

Issuer

Subject

Add Cancel

## 4. Basic Kaspersky SD-WAN configuration

### 4.1. Creating domains and data centers

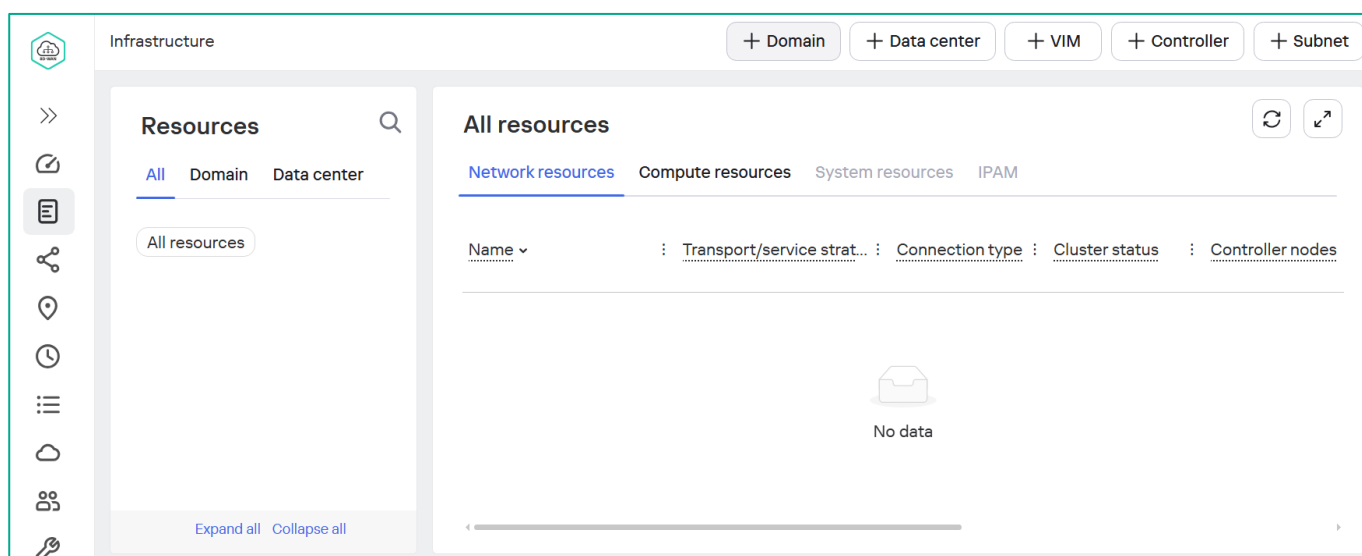
The orchestrator manages network and computing resources that may belong to different Domains and Data Centers.

A Domain is a logical group of resources under a single administrative management.

A Data Center is a logical entity that allows grouping of network and computing resources.

#### 4.1.1. Create a domain.

Click **+ Domain** in **Infrastructure** menu.



Enter domain **Name**.

Click **Create**.

New domain

Name

demolab.space

Description

Create

Cancel

## 4.1.2. Create a Data Center.

Click the **+ Data center** in **Infrastructure** menu.

Set data center parameters:

- Data center **Name**.
- **VNF URL**: <https://vnfm-proxy:86>

Click **Test Connection** (the connection test must be successful) and **Create**.

### New data center

Name

DC

Description

Domain

demolab.space

VNFM URL

https://vnfm-proxy:86

Test connection

Successful

Location

Moscow Olimpia Park

Бизнес-центр "Олимпия Парк", 39А с2, Leningrad Avenue, Voykovsky District, Moscow, Central Federal District, 125212, Russia

Add

Cancel



#### 4.1.3. Configure a connection to the Zabbix monitoring system.

Switch to the **System → Monitoring**.

Set Zabbix connection parameters:

- **Type: Zabbix.**
- **URL:** [https://zbx-www:8443/api\\_jsonrpc.php](https://zbx-www:8443/api_jsonrpc.php) (Zabbix API URL).
- **Login / Password:** use the login and password specified in step 3.4.2 (user to connect to the Zabbix API with read/write permissions in groups where CPEs are added for monitoring).
- **VNF/PNF Group: VNFGROUP** (Zabbix group where VNF/PNF will be added).
- **CPE Group: CPEGROUP** (Zabbix group where CPE will be added).

To connect to the Zabbix server, generate a token by clicking the **Generate** button.

Click **Test connection**, to verify if connection to the Zabbix server is OK (i.e., that the settings are correct).

Click **Apply**.

The screenshot shows the 'Monitoring' configuration page in the Kaspersky SD-WAN interface. The page has a sidebar with navigation icons and a main content area with tabs for 'Monitoring', 'Notification', 'Orchestrator log', 'User sessions', 'API', and 'Other'. The 'Monitoring' tab is active. The configuration fields are as follows:

Field	Value
Type	Zabbix
URL	https://zbx-www:8443/api_jsonrpc.pl
Login	Admin
Password	.....
Grouping by Zabbix	By specified groups
VNF/PNF group	VNFGROUP
CPE group	CPEGROUP
Trigger synchronization (sec)	600
Token	.....

At the bottom of the form, there are two buttons: 'Test connection' and 'Apply'. Below the 'Test connection' button, the status 'Successful' is displayed.

## 4.1.4. Configure data center system resources.

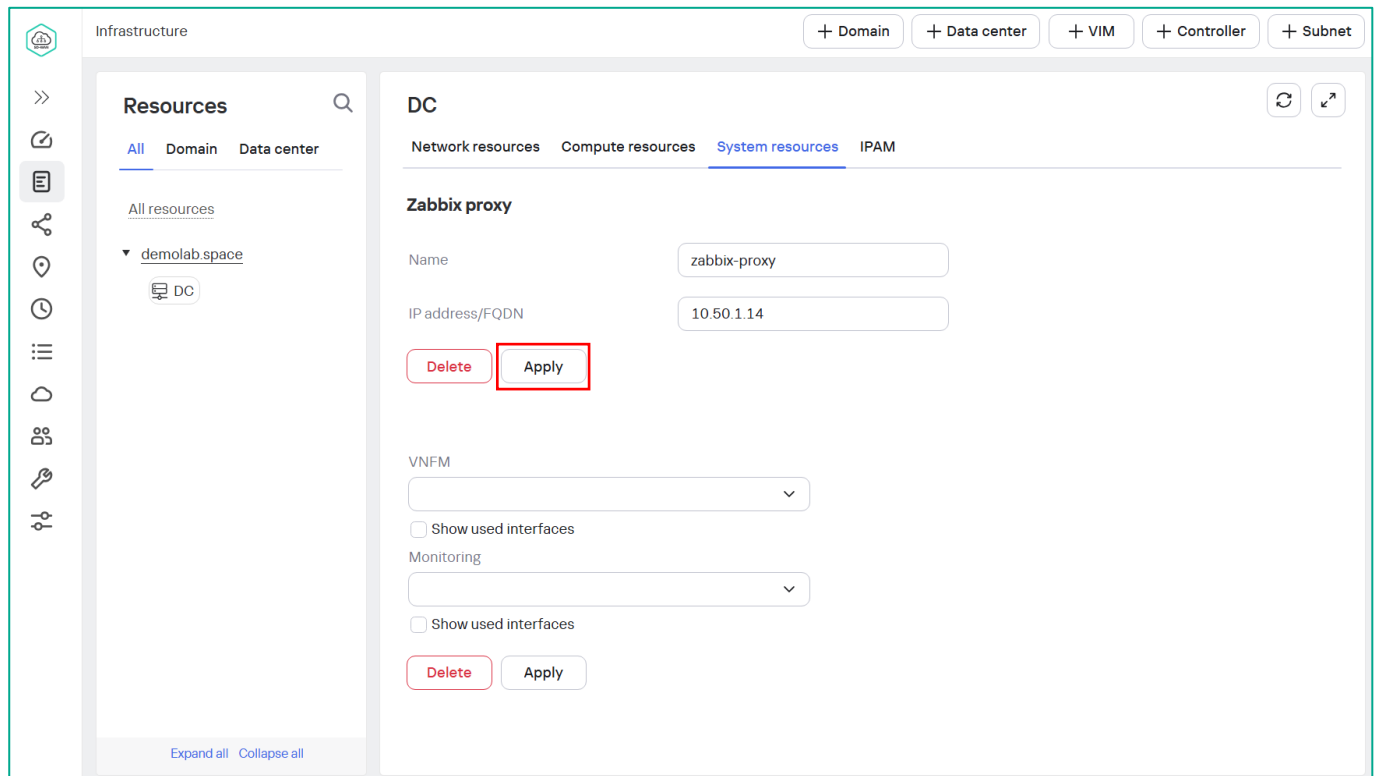
In the main menu on the left side select **Infrastructure**, then in the resources tree select **DC** and go to the **System Resources** tab.

Specify information for connecting to the Zabbix Proxy:

- **Name: zabbix-proxy** (must match the name specified in the Zabbix Server settings).
- **IP: 10.50.1.14** (public **orc1** IP-address).

**Note:** If you change the IP plan from 2.2, use the new IP address of the orc1 host.

Click **Apply**.



The screenshot shows the Kaspersky SD-WAN Infrastructure configuration interface. On the left, the 'Resources' tree is expanded to show 'demolab.space' > 'DC'. The main panel is titled 'DC' and has tabs for 'Network resources', 'Compute resources', 'System resources' (selected), and 'IPAM'. Under 'System resources', there is a 'Zabbix proxy' section. It contains two input fields: 'Name' with the value 'zabbix-proxy' and 'IP address/FQDN' with the value '10.50.1.14'. Below these fields are 'Delete' and 'Apply' buttons, with the 'Apply' button highlighted by a red rectangle. Further down, there is a 'VNFM' dropdown menu, a 'Show used interfaces' checkbox, and a 'Monitoring' dropdown menu, followed by another 'Delete' and 'Apply' button pair.

4.1.5. Add an IP address pool for the management network.

One or more management IP pools are allocated for each Data center.

Go to **Infrastructure → Domain → DC → IPAM** and click the **+ Subnet**

Specify the management subnet parameters:

- **Name:** mgmt.
- **CIDR:** 10.11.13.0/24.
- **IP Range:** 10.11.13.13 – 10.11.13.253 (to add a new range, click **+ Add**).

Click **Create**.

### New subnet ×

Domain  
demolab.space ▼

Data center  
DC ▼

Name  
mgmt

Type  
Management ▼

IP version  
IPv4 ▼

CIDR  
10.11.13.0/24

Gateway

IP range  
10.11.13.13 10.11.13.253 ×

+ Add

Create Cancel

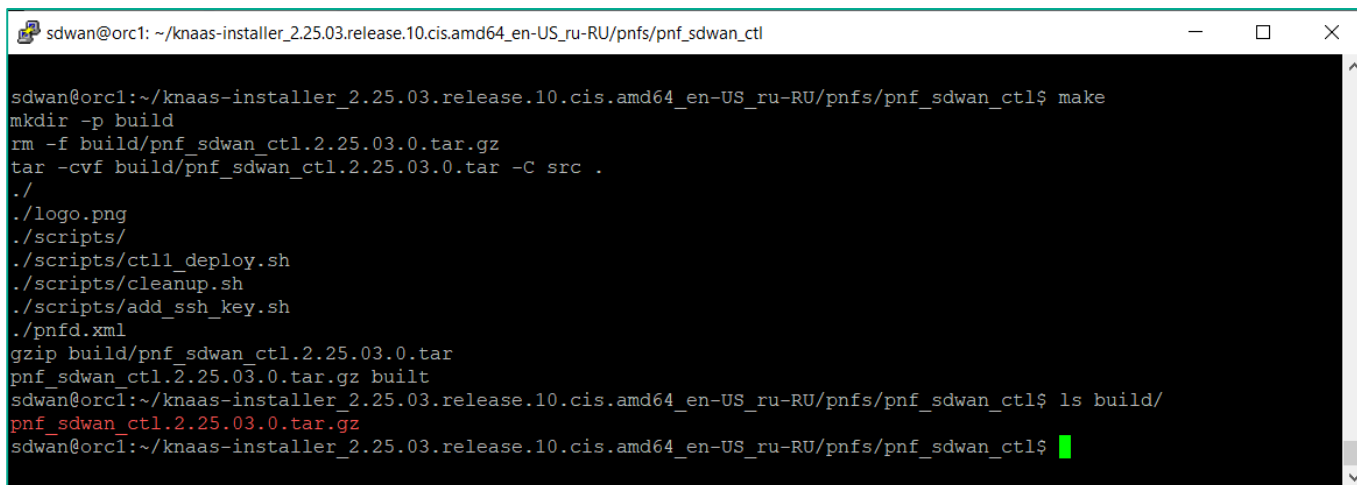
## 4.1.6. Create the SD-WAN controller PNF descriptor.

A sample PNF descriptor is located in the installation playbook archive at the following path:  
**/home/sdwan/knaas-installer.<release\_name>.gbl.amd64\_en-US\_ru-RU/pnfs/pnf\_sdwan\_ctl/src**

Run **make** from the following directory to create the PNF descriptor archive:

**/home/sdwan/knaas-installer.<release\_name>.gbl.amd64\_en-US\_ru-RU/pnfs/pnf\_sdwan\_ctl/**

The archive is created in the following path: **/home/sdwan/knaas-installer.<release\_name>.gbl.amd64\_en-US\_ru-RU/pnfs/pnf\_sdwan\_ctl/build/pnf\_sdwan\_ctl.2.25.03.0.tar.gz**



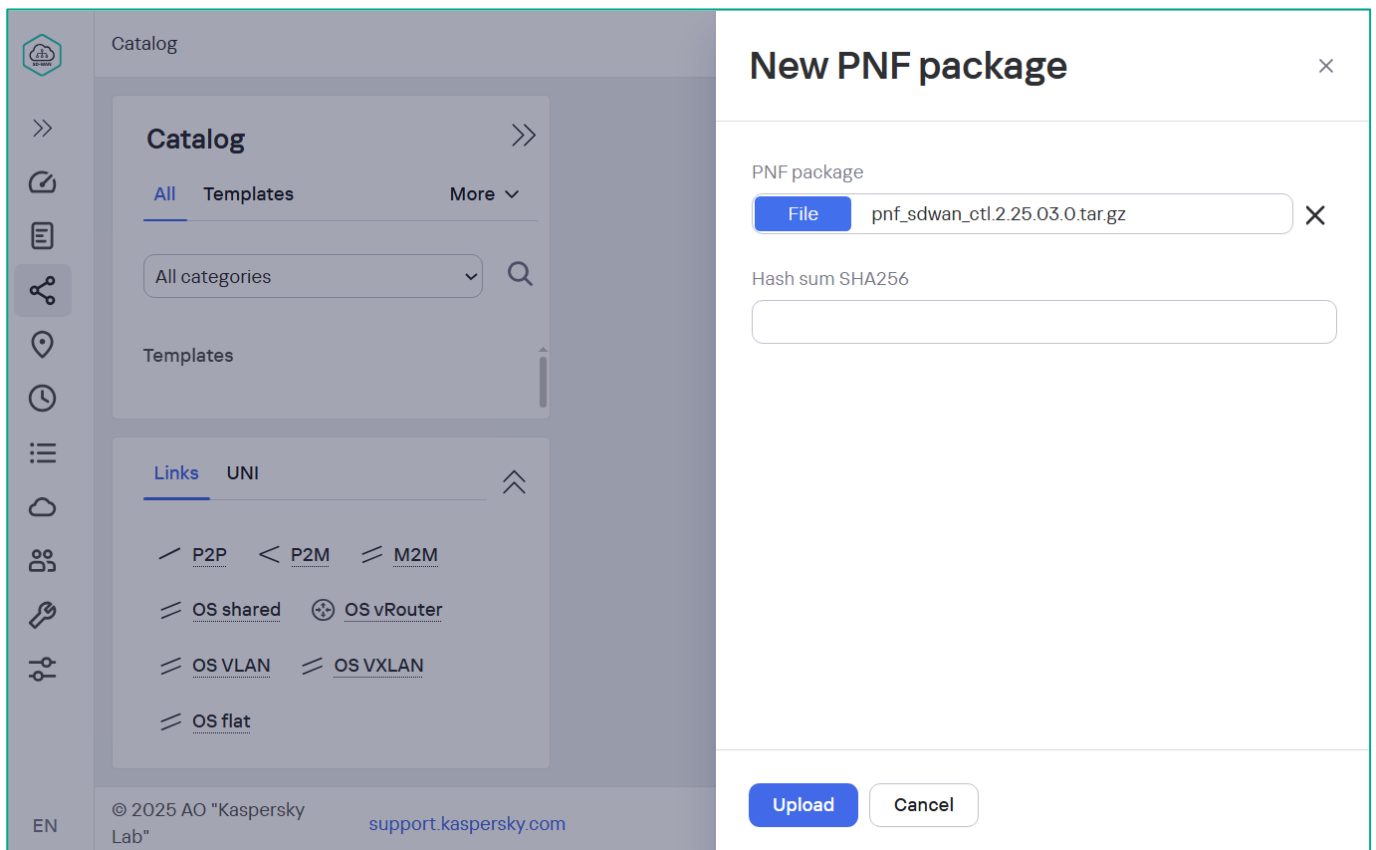
```
sdwan@orc1: ~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU/pnfs/pnf_sdwan_ctl
sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU/pnfs/pnf_sdwan_ctl$ make
mkdir -p build
rm -f build/pnf_sdwan_ctl.2.25.03.0.tar.gz
tar -cvf build/pnf_sdwan_ctl.2.25.03.0.tar -C src .
./
./logo.png
./scripts/
./scripts/ctl1_deploy.sh
./scripts/cleanup.sh
./scripts/add_ssh_key.sh
./pnfd.xml
gzip build/pnf_sdwan_ctl.2.25.03.0.tar
pnf_sdwan_ctl.2.25.03.0.tar.gz built
sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU/pnfs/pnf_sdwan_ctl$ ls build/
pnf_sdwan_ctl.2.25.03.0.tar.gz
sdwan@orc1:~/knaas-installer_2.25.03.release.10.cis.amd64_en-US_ru-RU/pnfs/pnf_sdwan_ctl$
```

Copy the descriptor archive from the **orc1** host.

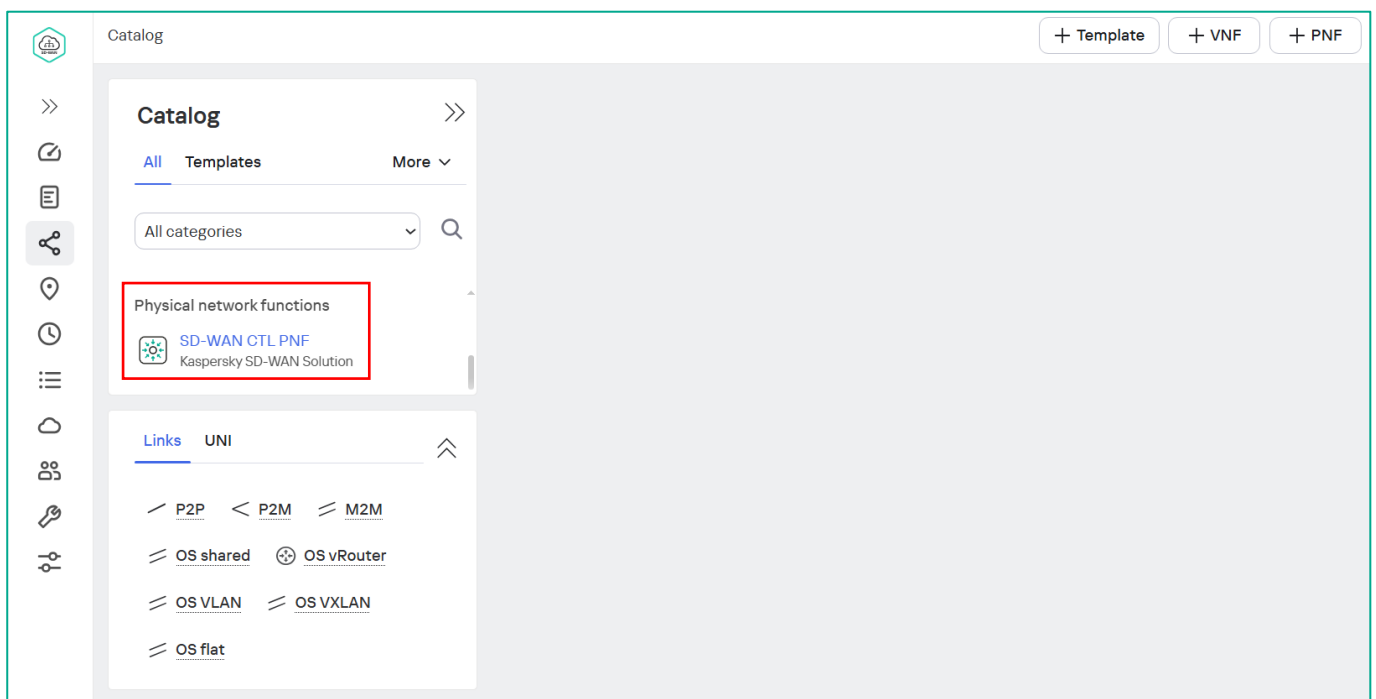
#### 4.1.7. Import PNF CTL to Catalog:

Go to **Catalog**, click **+ PNF** to add a new PNF package.

Specify the path and select **pnf\_sdwan\_ctl.2.25.03.0.tar.gz**, archive, then click **Upload**.



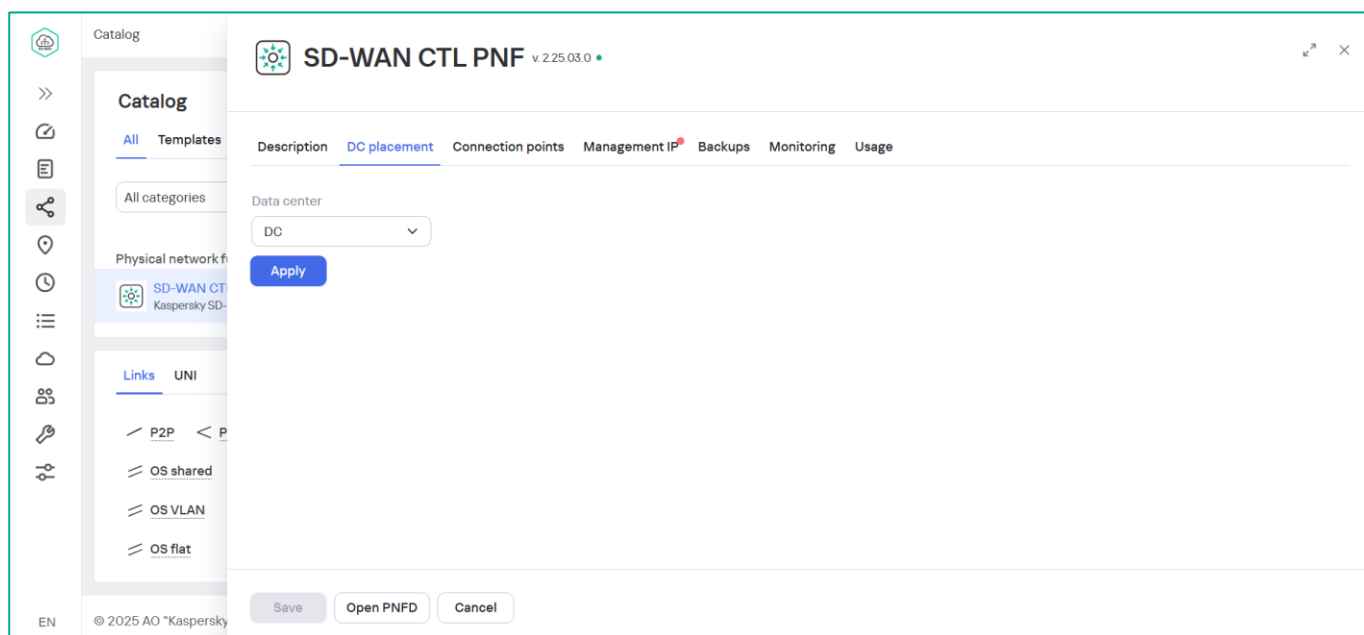
Wait for the PNF to be loaded into the catalog.



4.1.8. Set Data center for the PNF.

Select SD-WAN controller **PNF**: click **Physical Network Function** → **SD-WAN-CTL-PNF**.

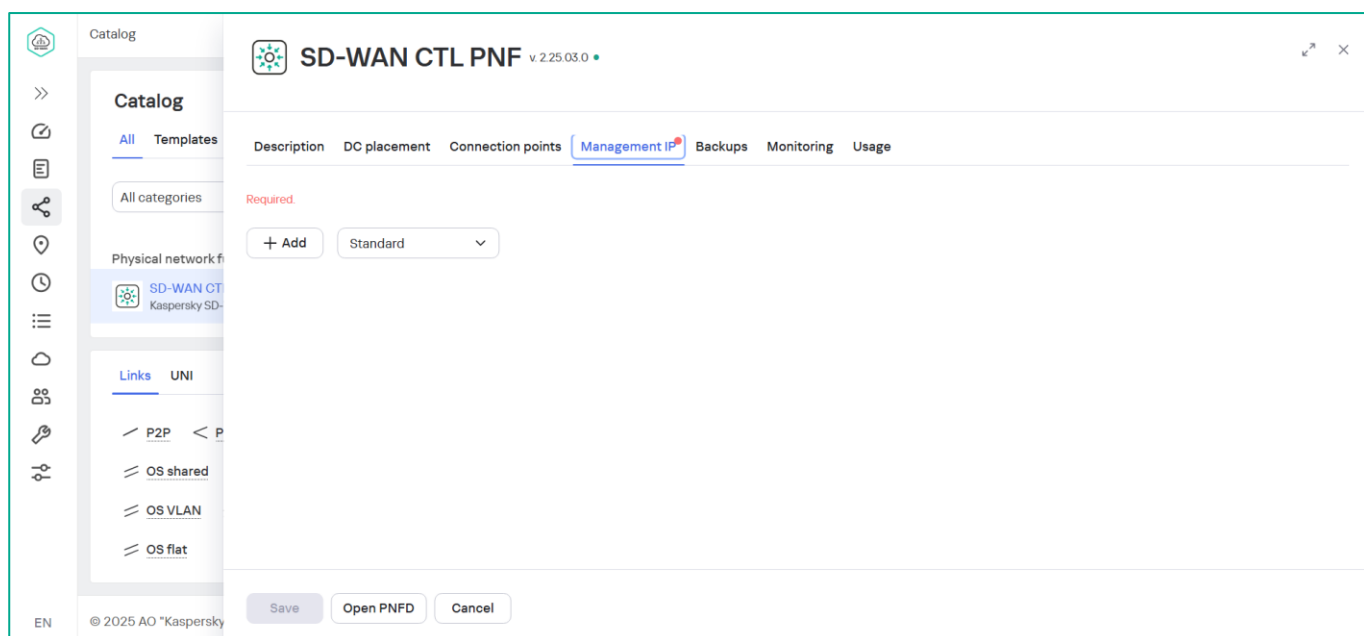
Switch to the **DC Placement** tab, select **DC**, click **Apply**.



4.1.9. Set the IP address for connection to the controller.

Switch to the **Management IP** tab.

Click **+ Add** to set Flavor **Standard**.



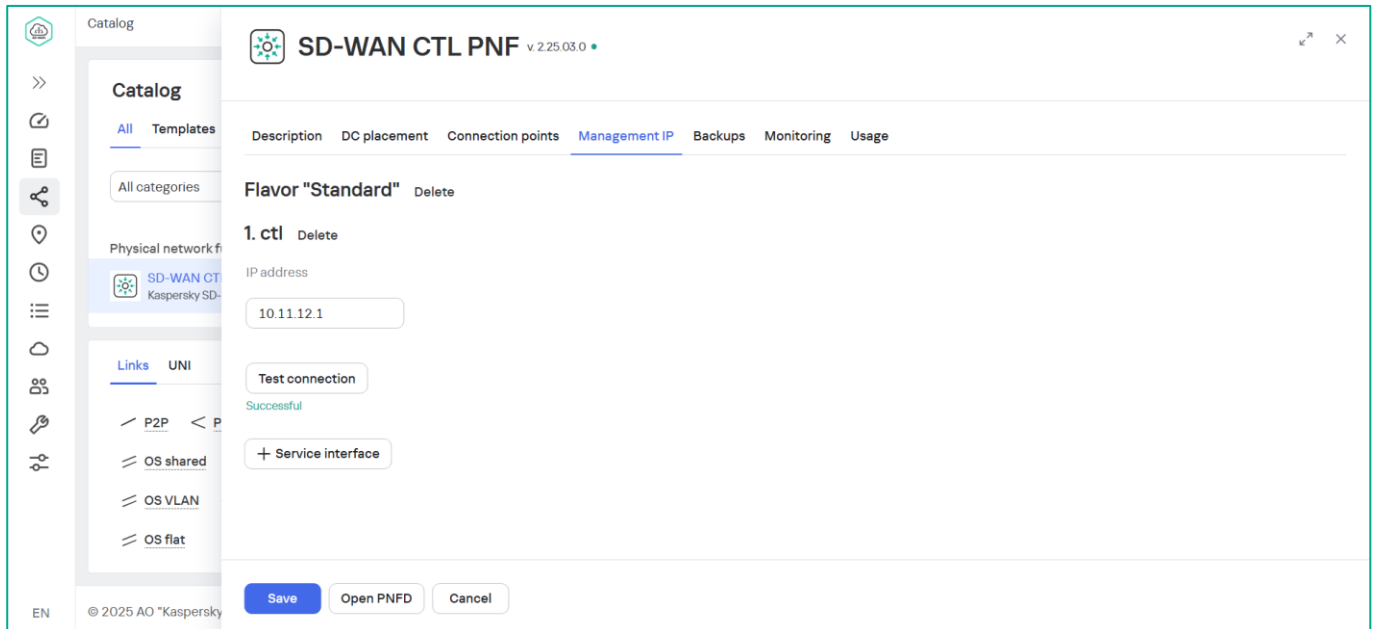
Click **+ Add** again to add the SD-WAN controller IP address.

Enter the IP address of the gateway from the Docker network **knaas\_os\_man: 10.11.12.1**.

**Note:** If this network is changed, use a new IP address for the controller connection.

Click **Test Connection** (test must be successful).

Click **Save**.

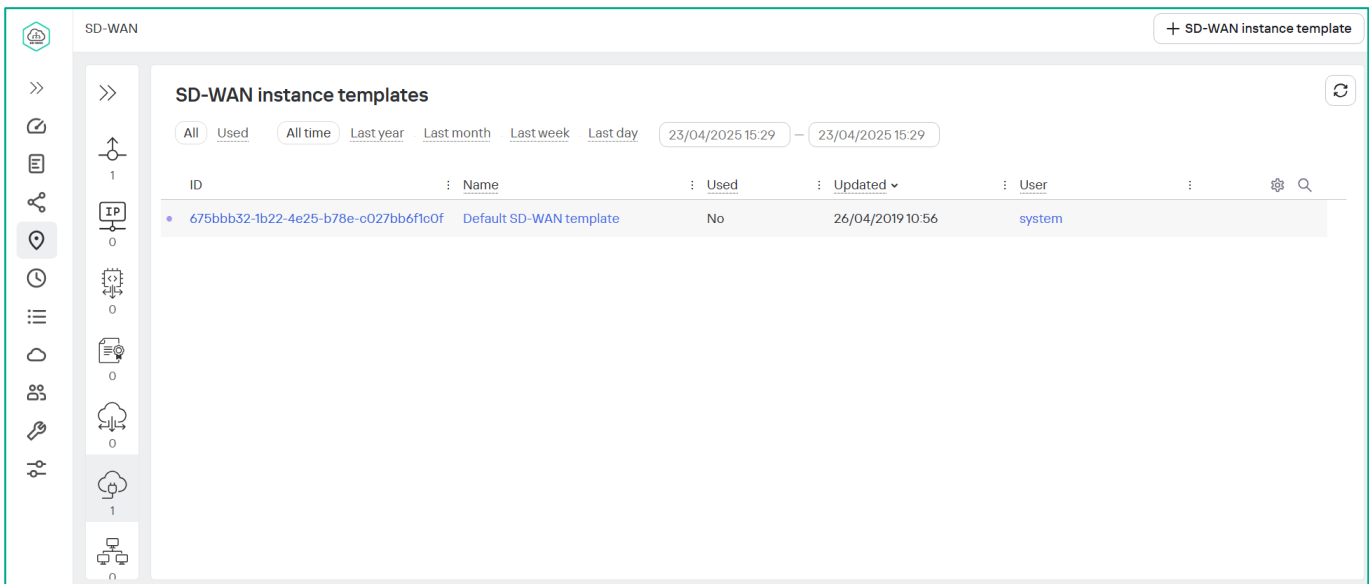


4.2. SD-WAN instance template configuration

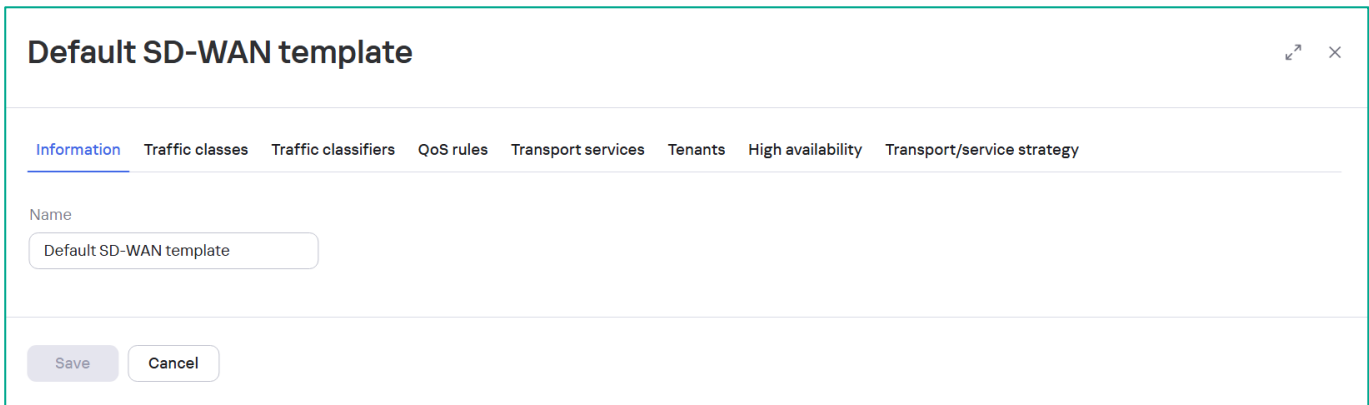
The SD-WAN Instance template contains parameters for an overlay network. After deploying the SD-WAN service, it applies to the SD-WAN controller.

4.2.1. Configure the default SD-WAN instance template.

Go to **SD-WAN → SD-WAN Instance templates** and click the **Default SD-WAN template** to edit it.



Change the template **Name** or use the default value.





4.2.2. Configure transport services in the SD-WAN instance template.

Switch to the **Transport Services** tab.

Delete the **SD-WAN P2M Data** service.

Data transport service creation is described in section 5.

Keep only the **SD-WAN management tunnel**.

The screenshot shows the 'Default SD-WAN template' configuration page with the 'Transport services' tab selected. The page has a top navigation bar with tabs: Information, Traffic classes, Traffic classifiers, QoS rules, Transport services (active), Tenants, High availability, and Transport/service strategy. Below this is a sub-navigation bar with 'X2M services' and 'L3 services'. A '+ Transport service' button is visible. A table lists the existing transport services:

Name	Type	Management transport service	Mode	MAC age (sec)	MAC learn mode	MAC table size	MAC table overload	Actions
SD-WAN managementTunnel	P2M	Yes	Classic	300	Learn and flood	2000	Flood	<a href="#">Edit</a> <a href="#">Delete</a>

At the bottom, there are 'Save' and 'Cancel' buttons.

4.2.3. Check the transport service settings for CPE management.

Click **Edit** to configure the SD-WAN **managementTunnel**.

The default values correspond to the information in the screenshot.

The screenshot shows the 'Default SD-WAN template' configuration page with the 'Transport services' tab selected. The 'Transport service' settings dialog is open, showing the configuration for the 'SD-WAN managementTunnel' service. The dialog has the following fields:

- Name: SD-WAN managementTunnel
- Type: P2M (dropdown)
- MAC learn mode: Learn and flood (dropdown)
- MAC age (sec): 300 (spinner)
- Mode: Classic (dropdown)
- Management transport service: ☒
- MAC table size: 2000 (spinner)
- MAC table overload: Flood (dropdown)

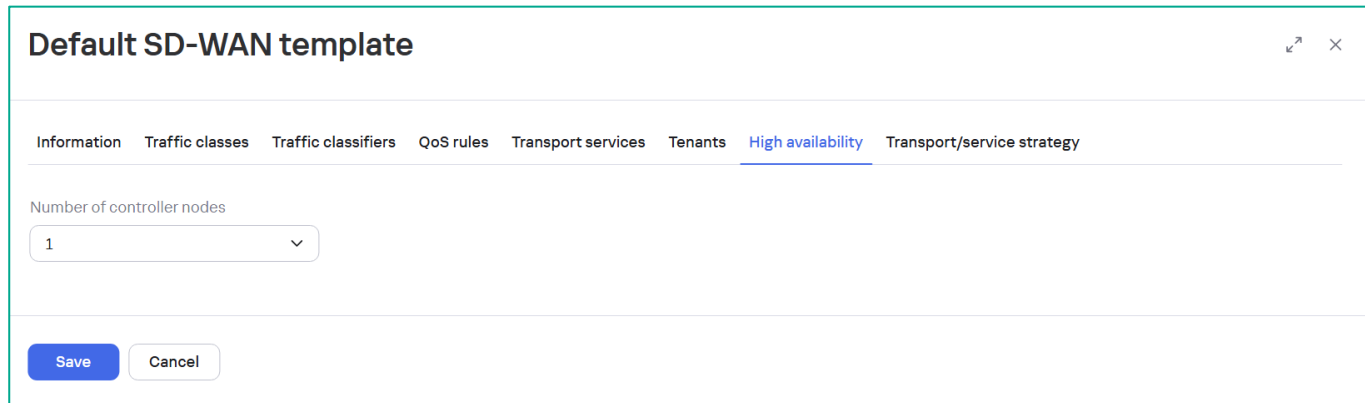
At the bottom of the dialog, there are 'Save' and 'Cancel' buttons.

4.2.4. Configure the number of controllers used by the SD-WAN service.

Switch to the **High Availability** tab.

Use default values:

- **Number of controller nodes: 1** (corresponds to the number of controllers in the PoC).



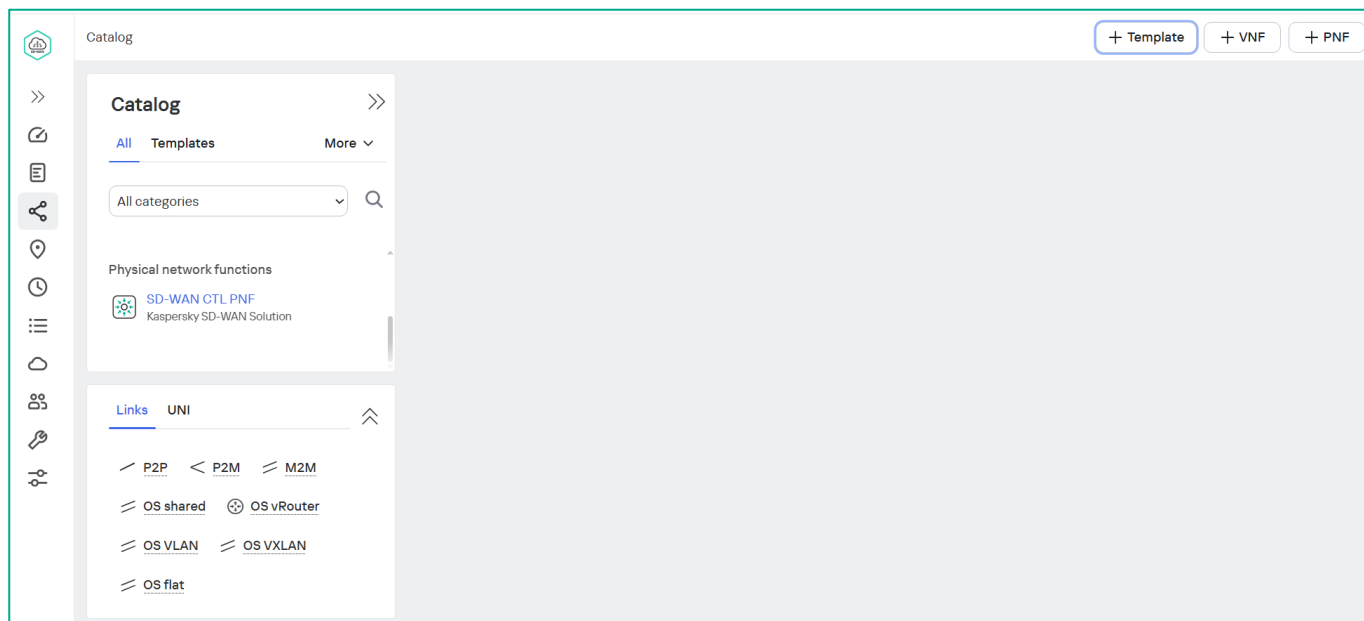
The screenshot shows a web interface titled "Default SD-WAN template" with a close button in the top right corner. Below the title is a horizontal tab bar with the following tabs: "Information", "Traffic classes", "Traffic classifiers", "QoS rules", "Transport services", "Tenants", "High availability" (which is selected and underlined), and "Transport/service strategy". Under the "High availability" tab, there is a label "Number of controller nodes" followed by a dropdown menu currently showing the value "1". At the bottom of the window, there are two buttons: "Save" (in blue) and "Cancel" (in light gray).

Click **Save** to apply changes.

## 4.3. Creating an SD-WAN service template

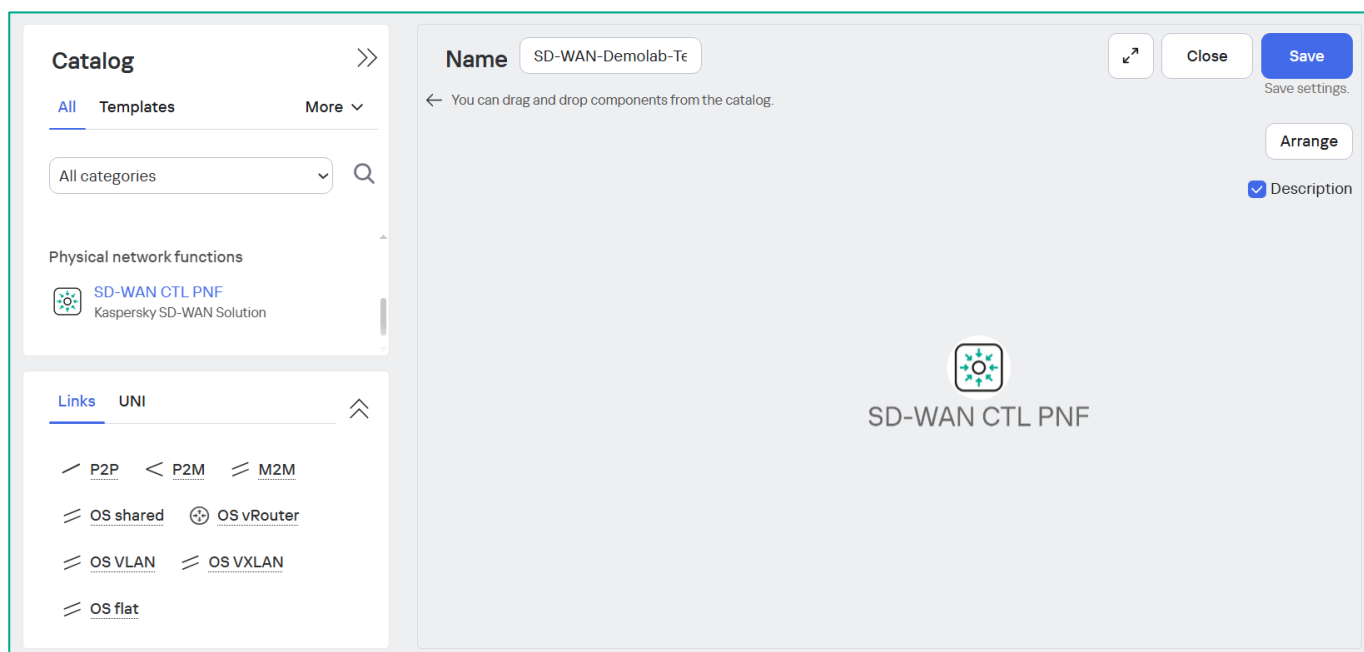
### 4.3.1. Create SD-WAN service template.

Go to **Catalog** menu. Click **+ Template** to add a new service template.



Drag and drop the PNF SD-WAN controller into the Designer window (**SD-WAN-CTL-PNF**).

Specify the **Name** of the template (in the example **SD-WAN-Demolab-Template**) and click **Save**.



4.3.2. Open the controller in the SD-WAN service template to edit parameters.

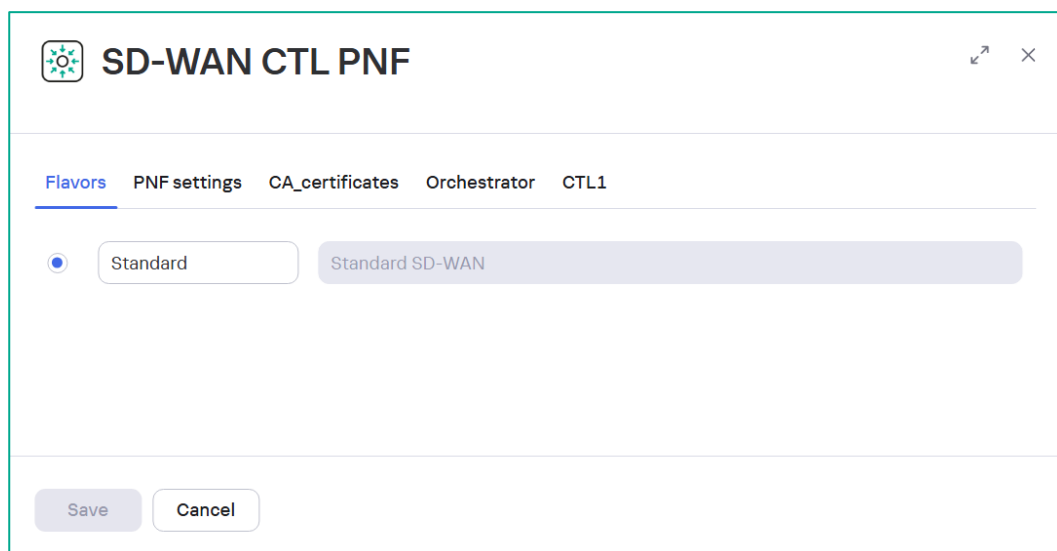
Select **SD-WAN-CTL-PNF** and click **Edit**.



Switch to the **PNF Settings** tab.

Set the SD-WAN controller **Name** or use the default value.

Click **Save**.



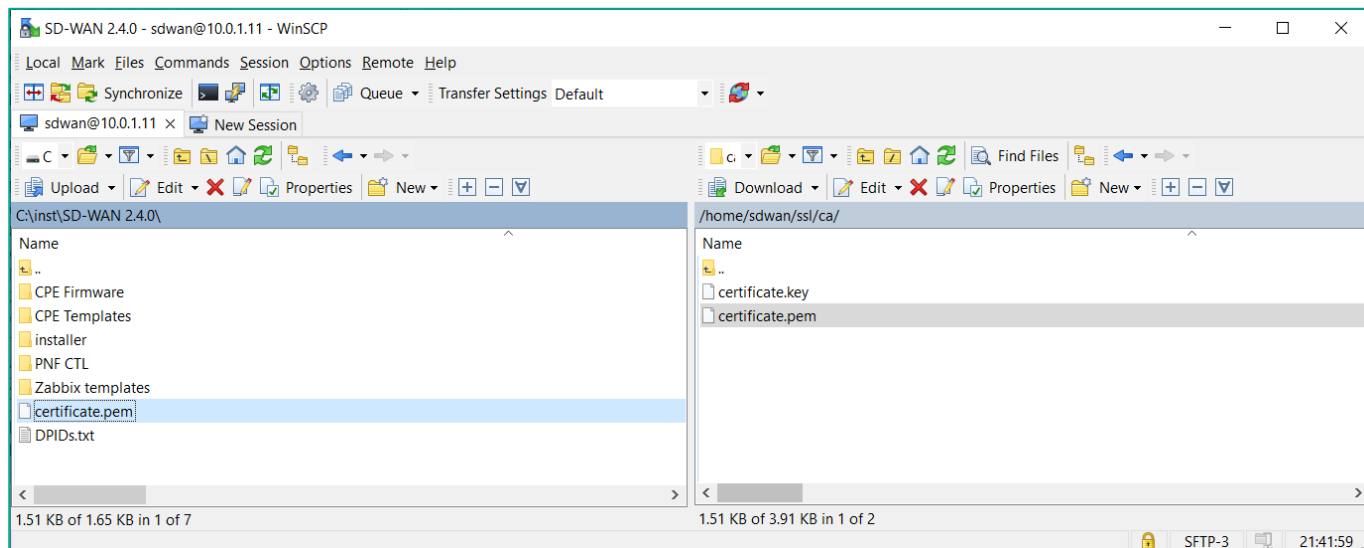
### 4.3.3. Add the orchestrator root certificate to the PNF of the controller.

When the controller connects to the orchestrator, certificate validation is performed, so you must add the root certificate that was used to sign the orchestrator certificate to the controller.

During the installation of the SD-WAN management system, the root CA certificate was saved to the file:

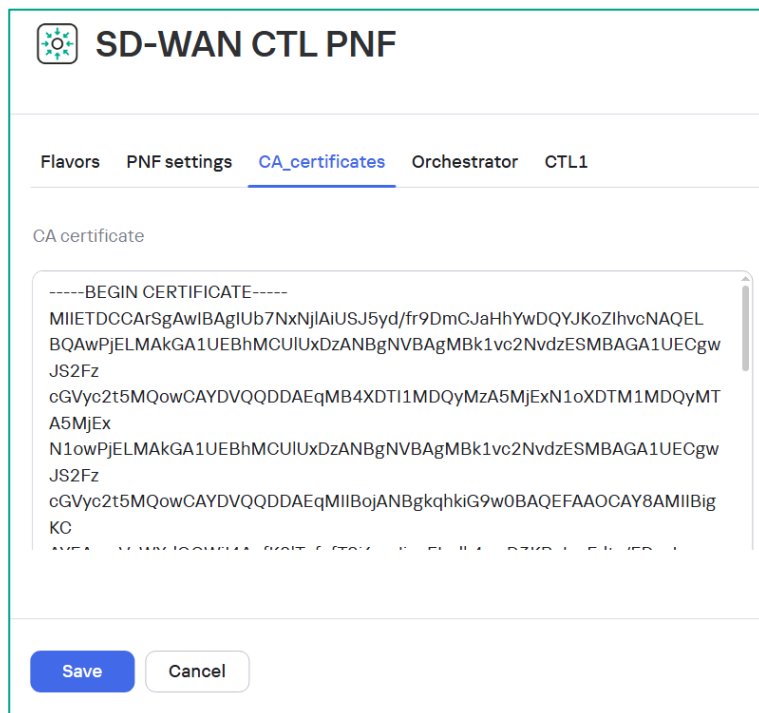
**/home/sdwan/ssl/ca/certificate.pem.**

Download a certificate from host orc1, e.g., using WinSCP.



Switch to the **CA\_certificates** tab.

Add the contents of the root certificate (**certificate.pem**) to the **CA certificate** in the plain text (the field can be expanded for easier viewing).

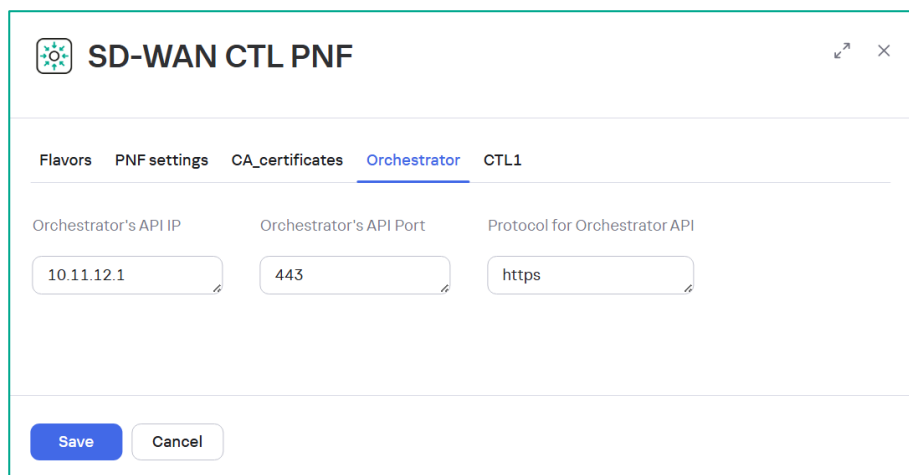


4.3.4. Set parameters for connecting the controller to the SD-WAN orchestrator.

Switch to the **Orchestrator** tab.

Enter the IP address of the gateway from the Docker network `knaas_os_man` (defined in step 3.2.6): **10.11.12.1**.

**Note:** If `knaas_os_man` network is changed, use a new orchestrator IP address.



The screenshot shows the 'SD-WAN CTL PNF' configuration window with the 'Orchestrator' tab selected. The configuration fields are as follows:

Field	Value
Orchestrator's API IP	10.11.12.1
Orchestrator's API Port	443
Protocol for Orchestrator API	https

At the bottom, there are 'Save' and 'Cancel' buttons.

4.3.5. Set parameters for connecting the orchestrator and CPE devices to the controller.

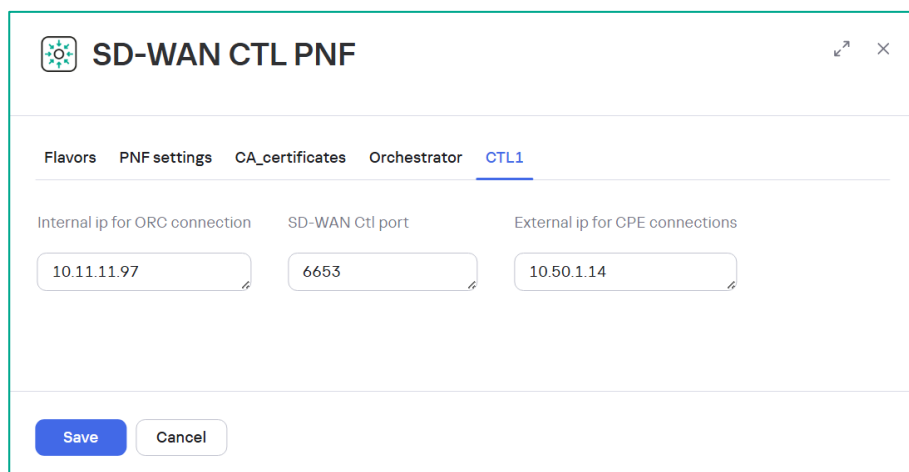
Switch to the **CTL1** tab.

As an internal IP address, use the IP address of the controller container: **10.11.11.97**

As an external IP address, use the public IP address of the controller: **10.50.1.14**, this address will be sent to the CPE for connection to the controller. On R14 DNAT is configured for ports 6653-6656 for this IP address.

**Note:** If you change the IP plan from 2.2, use the new public IP address of the `orc1` host.

Click **Save** in the controller settings.

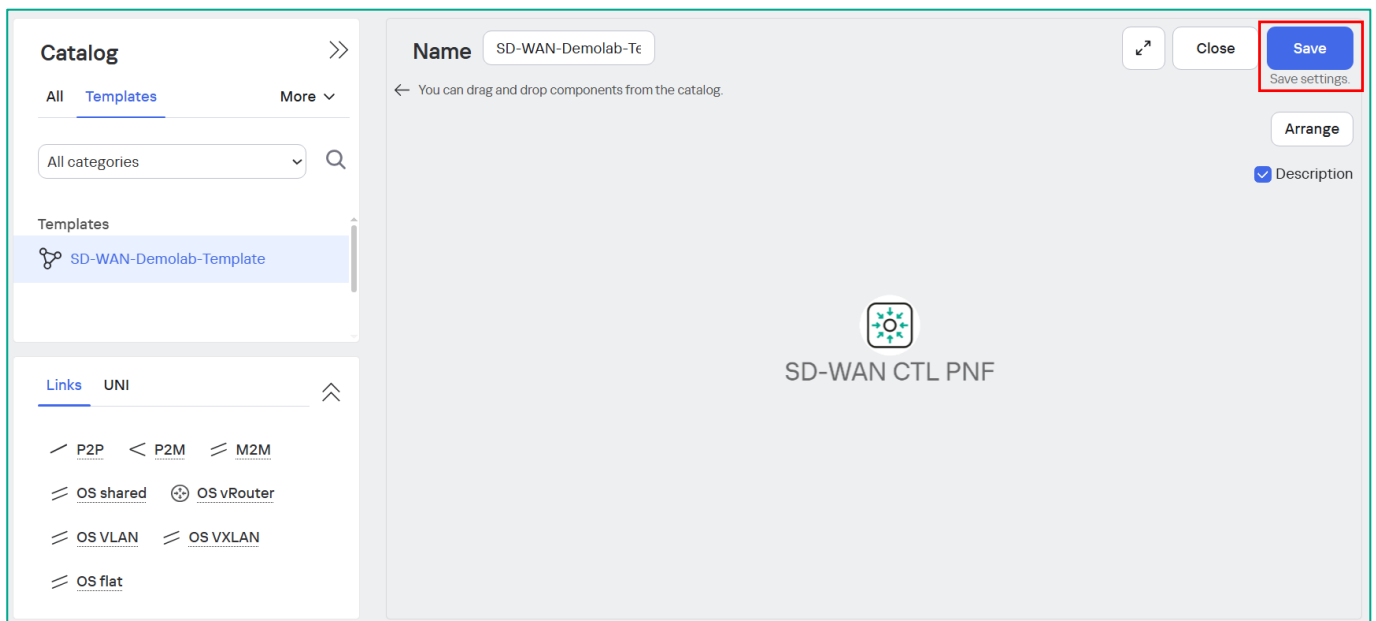


The screenshot shows the 'SD-WAN CTL PNF' configuration window with the 'CTL1' tab selected. The configuration fields are as follows:

Field	Value
Internal ip for ORC connection	10.11.11.97
SD-WAN Ctl port	6653
External ip for CPE connections	10.50.1.14

At the bottom, there are 'Save' and 'Cancel' buttons.

Then click **Save** in the template settings.



## 4.4. Creating a tenant and deploying an SD-WAN service

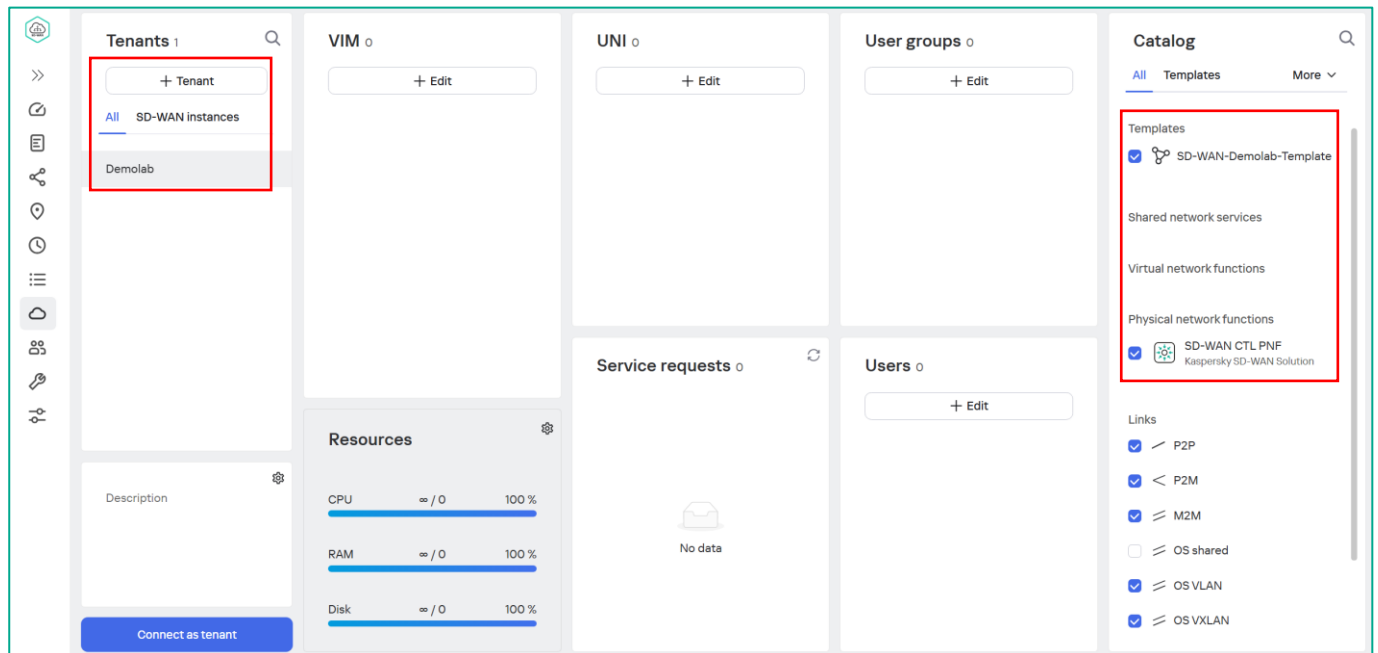
4.4.1. Create a new tenant.

Go to **Tenants** menu.

Click **+ Tenant**, enter new tenant name (**Demolab** in the example) and the click **+** to create a new tenant.

**Note:** Do not use "." or special characters in the name.

In the **Catalog** section check: **Templates** and **Physical Network Functions**. These objects will be available for tenant use.





#### 4.4.2. Add tenant administrator (Optionally).

Create new user with tenant role.

Go to **Users** menu. Click **+ Add**.

The screenshot displays the Kaspersky SD-WAN management interface. On the left is a sidebar with various icons, including a home icon, a list icon, a share icon, a location pin icon, a clock icon, a menu icon, a cloud icon, a group of people icon (highlighted), a key icon, and a settings icon. The main panel has a top navigation bar with tabs: **Users** (selected), **Permissions**, **Groups**, **LDAP connections**, and **Two-factor authentication**. Below the tabs is a large light gray button labeled **+ Add**. Underneath the button is a table with the following columns: **Name**, **Tenant**, **Role**, **Source**, **Group**, **State**, and **Two-factor authentication**. The table contains two rows of data:

Name	Tenant	Role	Source	Group	State	Two-factor authentication
Administrator Administrator		Administrator	Local	Default	Online	Disabled
User User		Tenant	Local	Default	Offline	Disabled

Specify new user parameters:

- **Login.**
- **Password.**
- **Role: Tenant.**
- **Permissions: Full access.**
- **First name / Last name.**

Click **Create**.

### New user

×

Source

Local

▼

Login

demolab-admin

Password

.....

👁

Password confirmation

.....

👁

Role

Tenant

▼

Permissions

Full access

▼

Two-factor authentication

☐ Off

Request confirmation is required

☐ Off

First name

Tenant

Last name

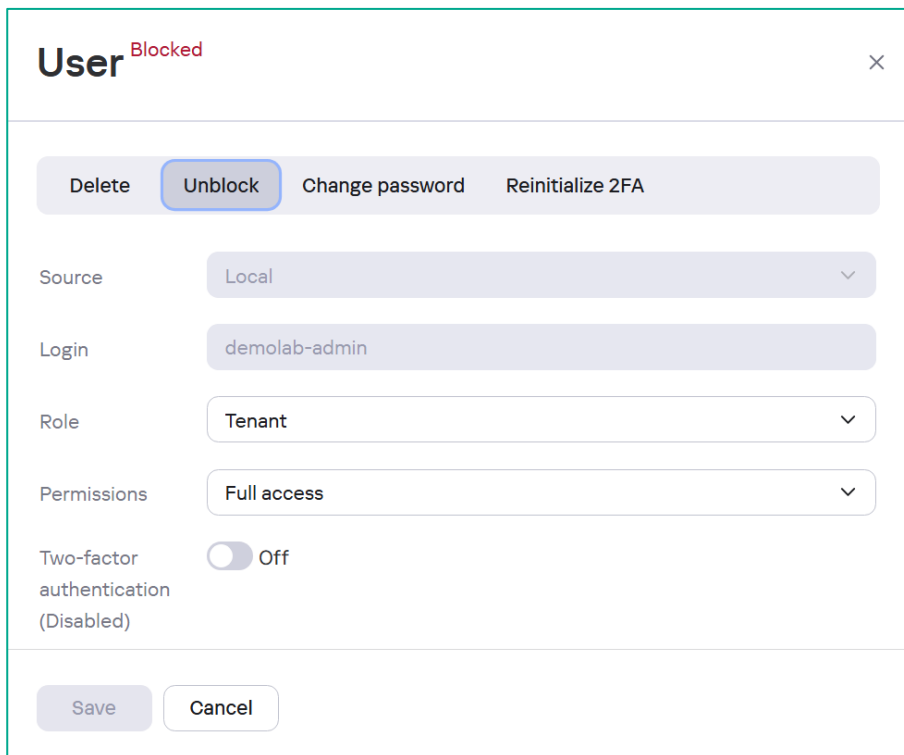
Tenant

Create

Cancel

#### 4.4.3. Activate a new user.

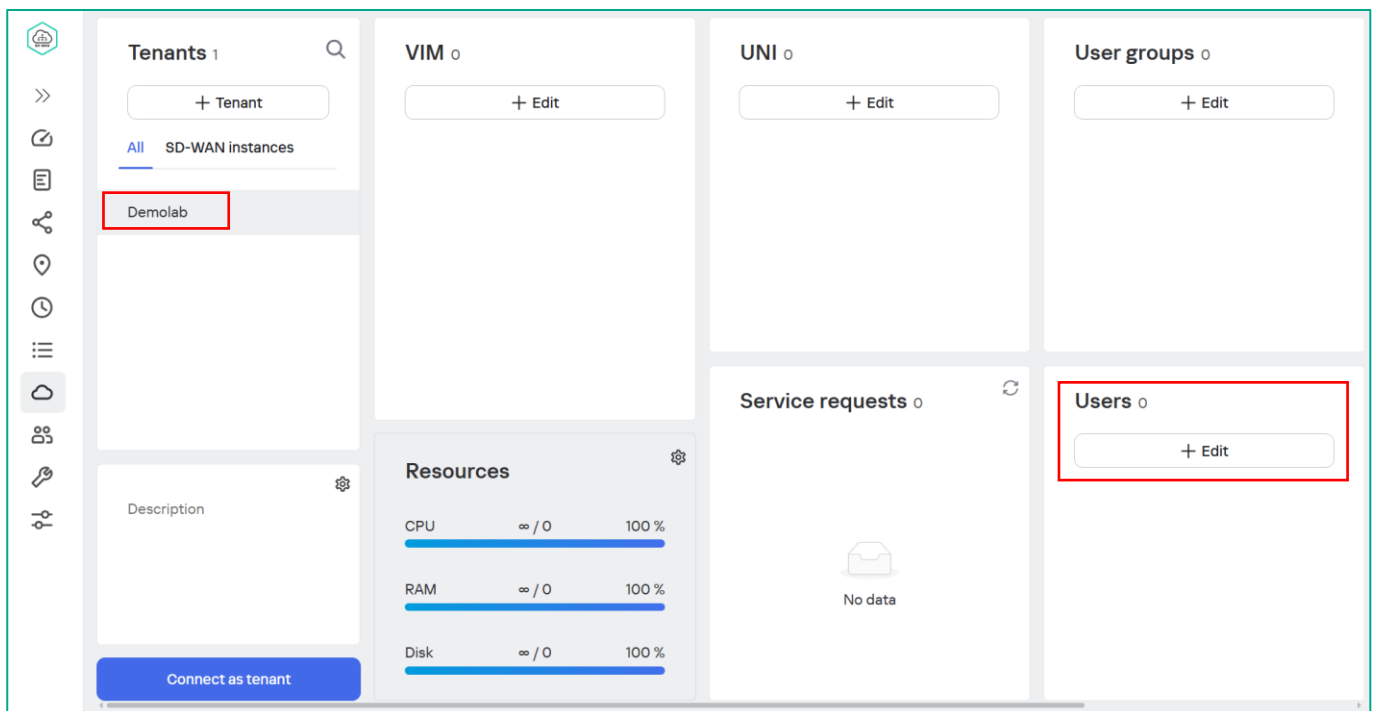
Select the user, you want to unblock, then click **Unblock**.



The image shows a 'User' management modal window. At the top, the word 'User' is followed by 'Blocked' in red. Below this is a toolbar with four buttons: 'Delete', 'Unblock' (which is highlighted with a blue border), 'Change password', and 'Reinitialize 2FA'. The modal contains several fields: 'Source' (set to 'Local'), 'Login' (set to 'demolab-admin'), 'Role' (set to 'Tenant'), and 'Permissions' (set to 'Full access'). There is also a 'Two-factor authentication' toggle switch, which is currently 'Off' and labeled '(Disabled)'. At the bottom of the modal are 'Save' and 'Cancel' buttons.

#### 4.4.4. Add the user to the previously created tenant.

Go to **Tenants** menu. Select the previously created tenant and click **+ Edit** in **Users** section.



The image is a screenshot of the Kaspersky SD-WAN management interface. On the left is a sidebar with various icons. The main area is divided into several panels. The 'Tenants' panel shows a list with 'Demolab' selected and highlighted with a red box. Below it is a 'Description' field and a 'Connect as tenant' button. The 'VIM' panel has a '+ Edit' button. The 'UNI' panel has a '+ Edit' button. The 'User groups' panel has a '+ Edit' button. The 'Service requests' panel shows 'No data' with a folder icon. The 'Resources' panel displays CPU, RAM, and Disk usage, all at 100%. The 'Users' panel, located at the bottom right, has a '+ Edit' button highlighted with a red box.

Move user to the **Assign users** and click **Save**.

Tenant's users

Users

User User

Assign users

< Tenant Tenant

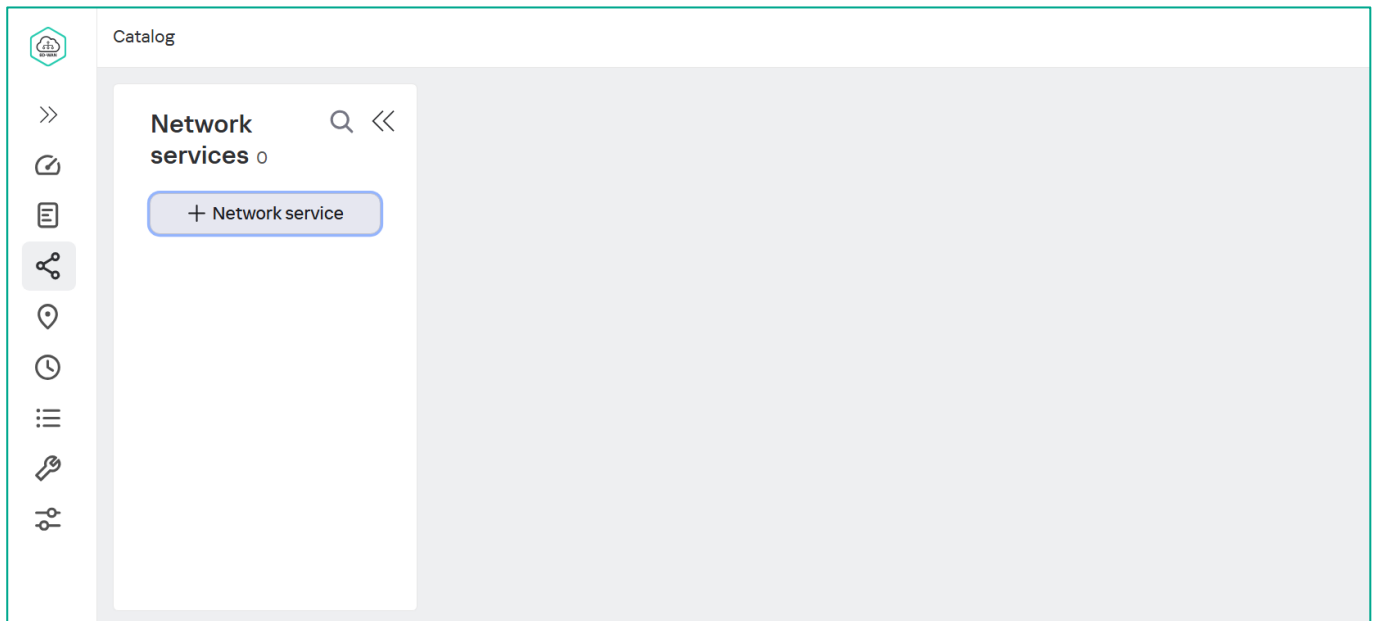
Save

Cancel

4.4.5. Deploy the SD-WAN network service from the SD-WAN service template.

Click **Connect as Tenant** or connect to the SD-WAN orchestrator by the Tenant administrator.

In the **Catalog** click **+ Network service**.

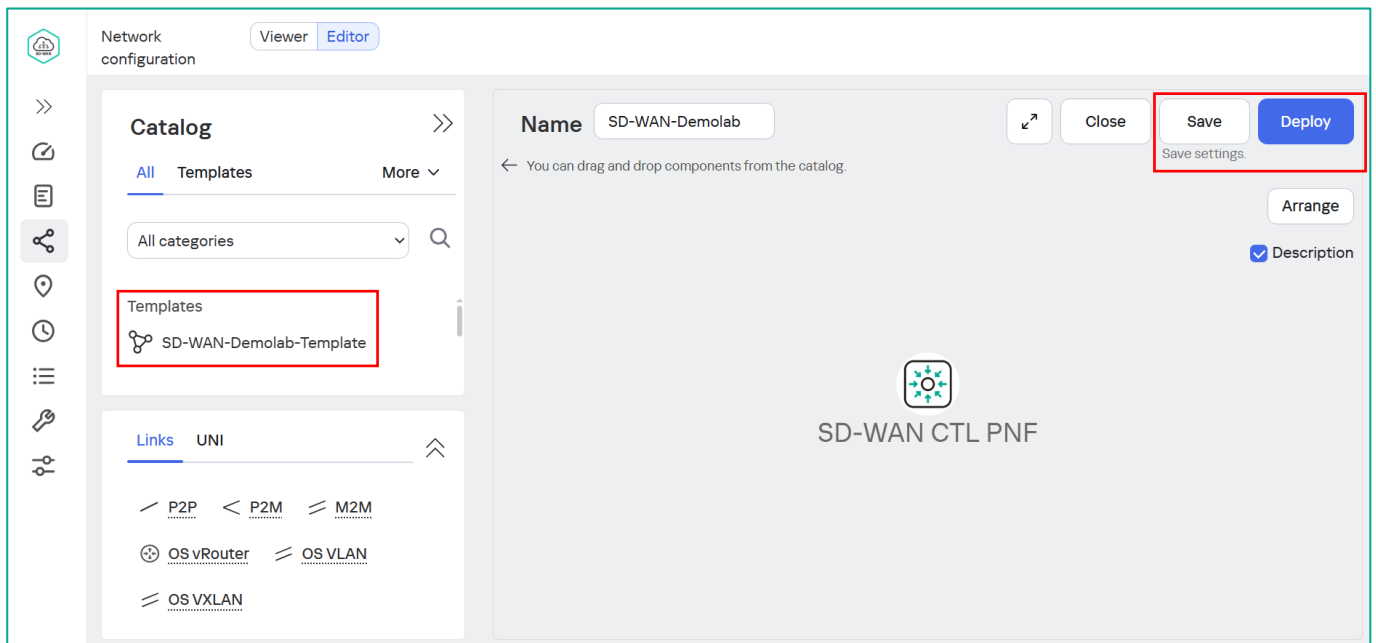


In the **Templates** section, select the previously created SD-WAN template and drag it to the Service Builder window.

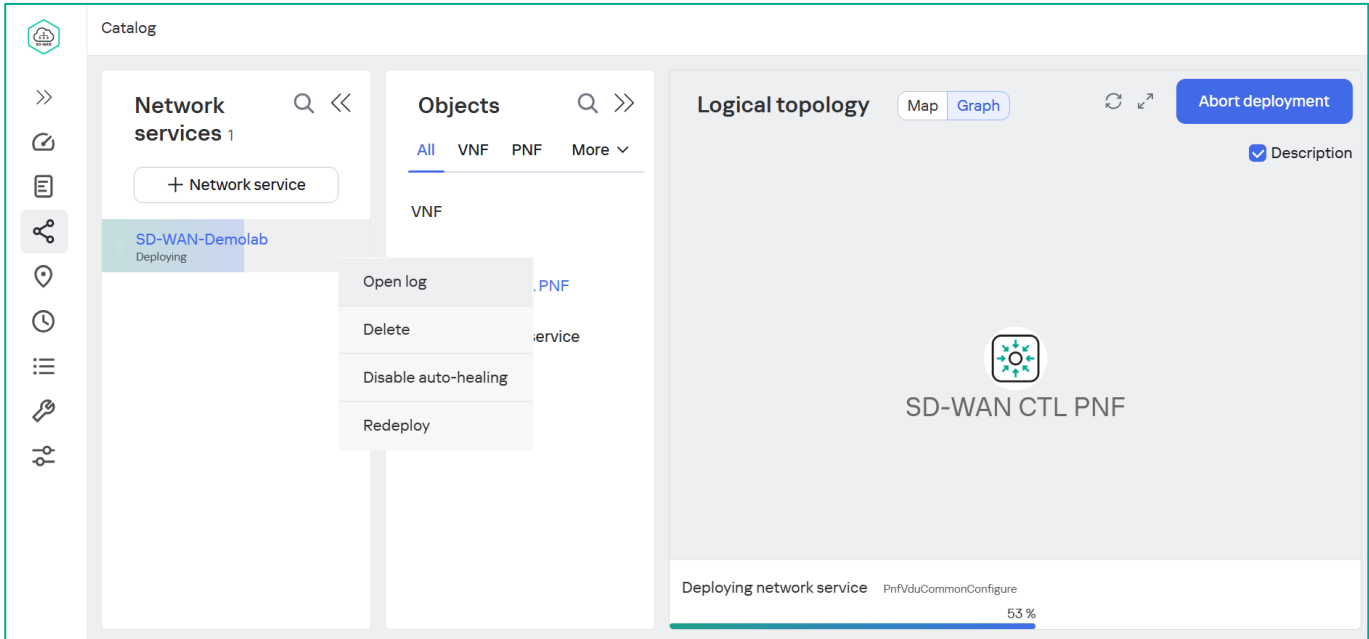
Set the **Name** of the SD-WAN service (**SD-WAN-Demolab** in the example).

Click **Save**.

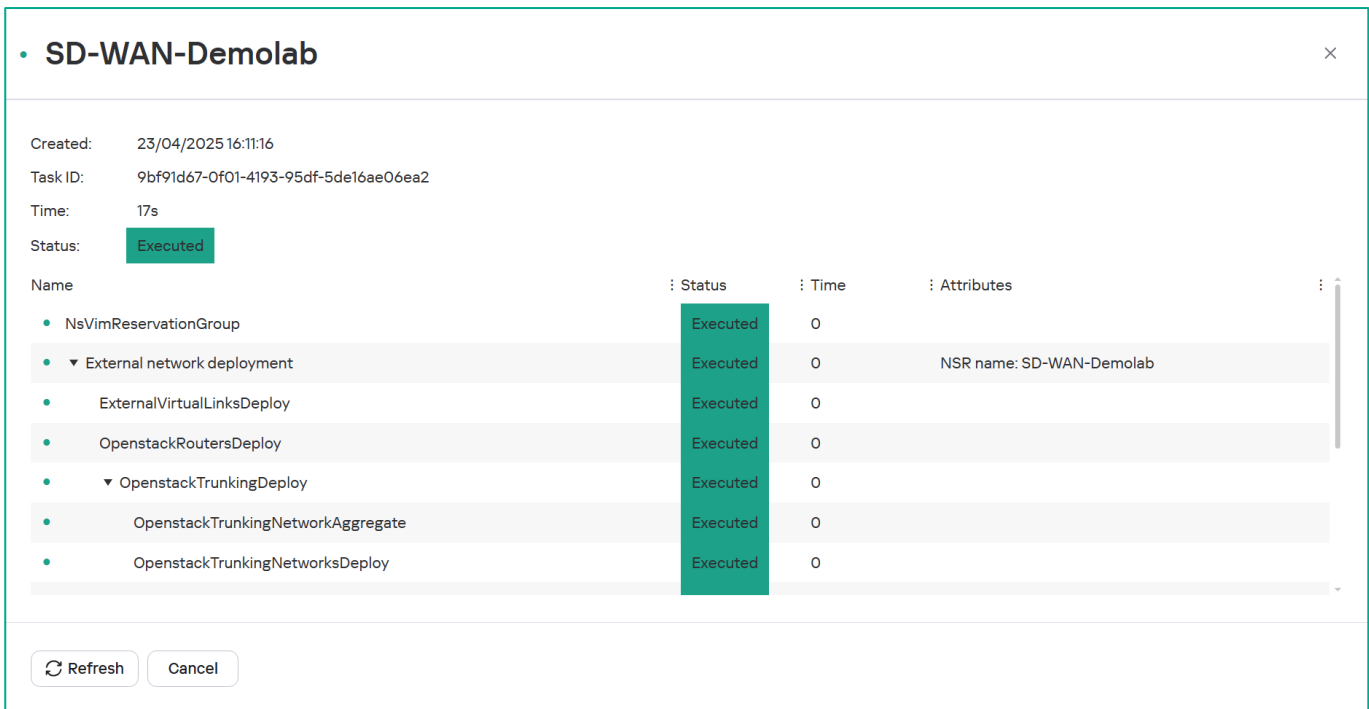
Click **Deploy**.



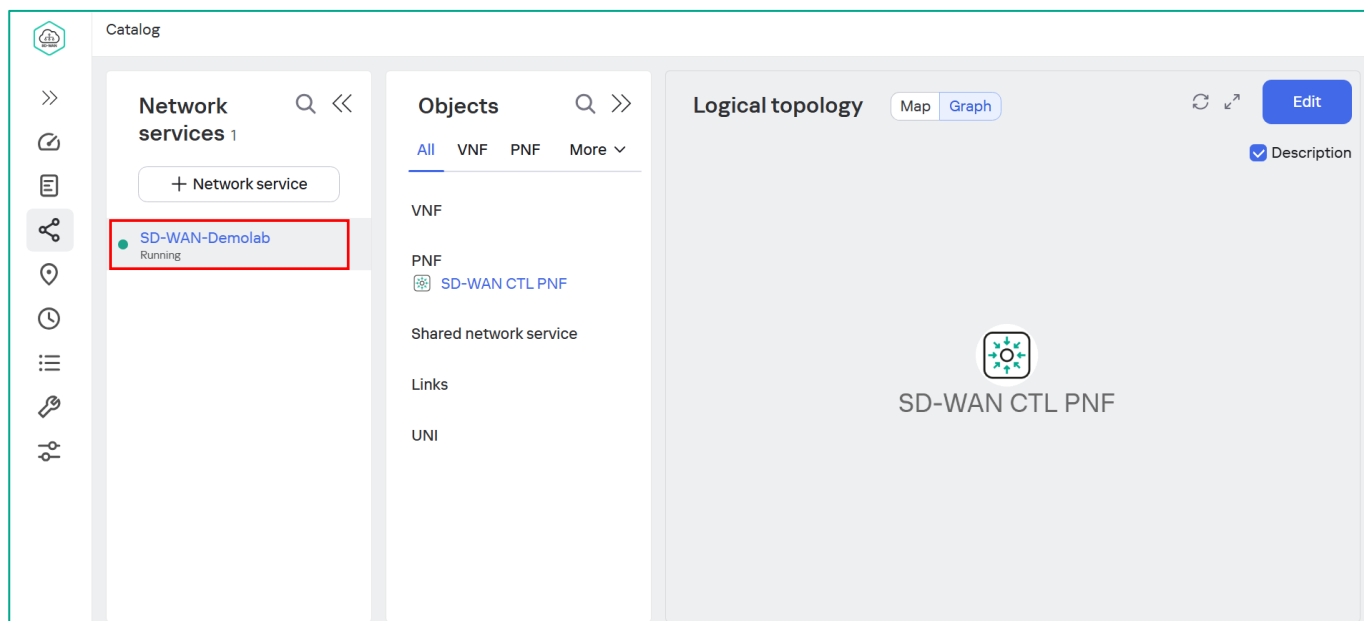
To monitor the deployment status of the service components, click the service configuration button (with the gear icon) and select **Open log**.



The interface displays the service deployment process with detailed status by individual task.



Wait until the deployment of the SD-WAN service has finished. In the **Network services** area, the SD-WAN service should be marked with a green icon with the status **Running**.



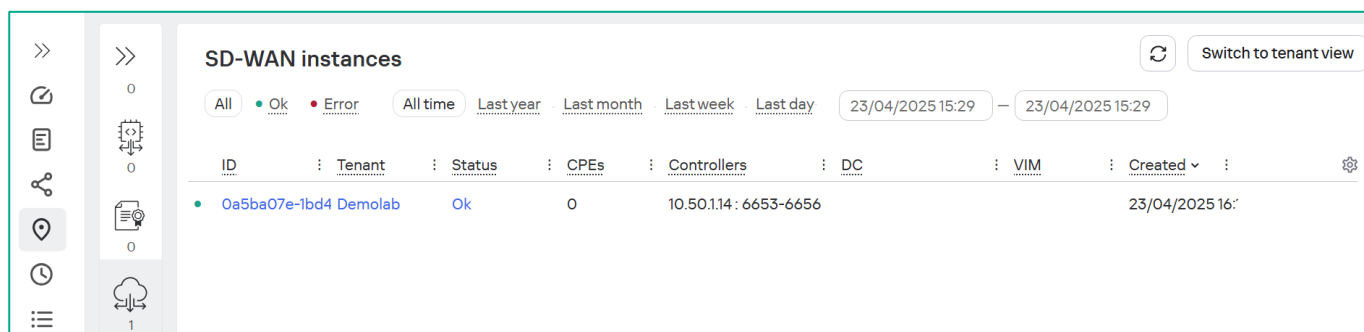
#### 4.4.6. Verify SD-WAN Service deployment

After successful SD-WAN service deployment on the Tenant, it is necessary to make sure that the service deployment has been successfully completed

Connect to the orchestrator as an administrator: repeat 3.3 or return to the previous browser window.

Go to **SD-WAN → SD-WAN Instances**.

A green icon should be displayed in front of the SD-WAN service **ID**.



4.4.7. Verify controller cluster status.

Go to **Infrastructure → Domain → DC → Network resources**.

Check the SD-WAN controller status. **Cluster status - Degraded** means the controller deployed without fault tolerance and operates in a single mode.

DC

Network resources

Compute resources

System resources

IPAM

Name	Transport/service strat...	Connection type	Cluster status	Controller nodes	Node statuses	Role in cluster	Active gRPC read node	
SD-WAN Cluster [Demolab: 4f7461d3-0a4b-4802-8b3e-8f9c8831f5ca]	Generic VNI swapping transport	Unicast	Degraded	10.11.11.97	Connected	Single	No	Management

4.4.8. Stop the temporary mockpnf container.

When deploying the SD-WAN service and configuring the controller, the orchestrator connects to the controller through the **mockpnf-1** container with a temporary password. After the service is deployed, the temporary container must be stopped.

Run the following command on the **orc1** host:

```
docker stop mockpnf-1
```



## 4.5. Creating firewall template for SD-WAN gateways

The SD-WAN firewall template includes parameters that are applied during registration or reboot of the device.

### 4.5.1. Create an additional firewall zone for gateways.

Connect to the tenant self-service portal (PoC uses the tenant **Demolab**), to do this, click **Connect as Tenant** from the **Tenants** menu or connect to the SD-WAN orchestrator by the administrator of the created tenant.

**Note:** When zones and firewall templates are created by an administrator from the administrator portal, they will not be available to users with tenant roles.

To enable SSH access to the CPE console from the orchestrator web interface to work, you must provide network connectivity between the orchestrator vnfm-1 container and the CPE IP addresses from the management network (mgmt). CPE interfaces on the management network are automatically added to a separate P2M transport service that has no connectivity to other networks, including the orchestrator subnet. To provide connectivity from the orchestrator to the CPE management addresses, you must configure masquerading for the zone which is assigned to the gateway management interfaces, so that the IP address of the orchestrator is translated to the gateway management interfaces (mgmt) IP addresses.

It is also possible to configure Source NAT in the firewall template from the IP address of the orchestrator to the IP addresses of the mgmt gateway interfaces, the assigned IP addresses can be viewed in **Infrastructure → DC → IPAM → Usage**.

Go to **SD-WAN → Firewall zones**.

Click **+ Firewall Zone**.

Name	Usage	Author	Created
lan	Yes	admin (Demolab)	23/04/2025 15:57:47
wan	Yes	admin (Demolab)	23/04/2025 15:57:47
mgmt	No	admin (Demolab)	23/04/2025 15:57:47

Enter the new firewall zone **Name: mgmt\_gw**.

Check **Masquerading**.

When forwarding packets from other zones to the mgmt\_gw zone, source network address translation (SNAT) is performed to the interface IP address from the mgmt\_gw zone.

New firewall zone

Name

mgmt\_gw

Input

ACCEPT

Output

ACCEPT

Forwarding

REJECT

☒ Masquerading

☒ MSS clamp to PMTU

☐ Drop logging

Masquerading source subnets

+ Add

Masquerading destination subnets

+ Add

Networks

+ Add

Create

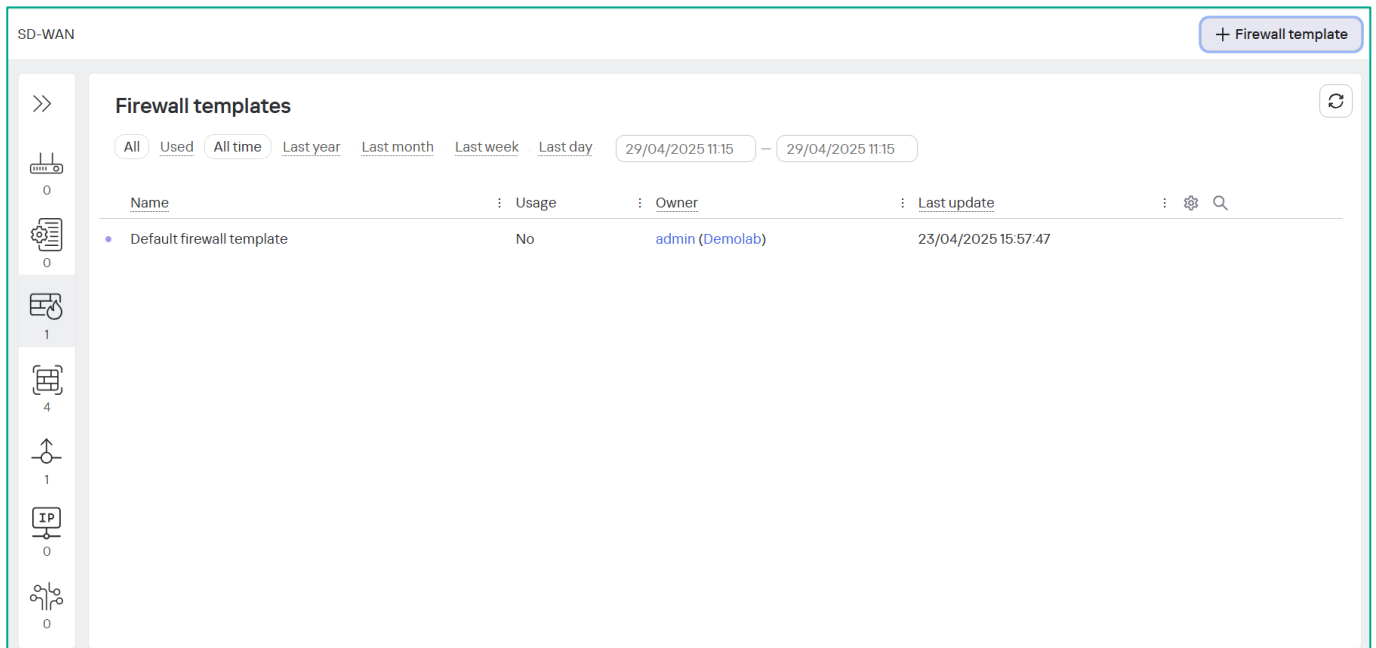
Cancel

Click **Create**.

#### 4.5.2. Create a gateway firewall template.

Go to **SD-WAN** → **Firewall templates**.

Click **+ Firewall Template**.



Enter the firewall template **Name: gateway\_firewall\_template**.

Click **Create**.

### New firewall template

Name

gateway\_firewall\_template

Create

Cancel

4.5.3. Configure zone forwarding rules for the SD-WAN Gateways firewall template.

Open the created template.

Switch to the **Zones forwarding** tab.

Create 2 forwarding rules: from zone **lan** to **mgmt** and from **mgmt** to **lan**.

Click **+ Forwarding** and select the required zones.

gateway\_firewall\_template

General settingsRulesNATZone forwardingIP address setsDPI marking

Set as designatedDeleteImportExportCloneShow associated CPEs

+ Forwarding

From	To	Actions
lan (Demolab)	mgmt_gw (Demolab)	Delete
mgmt_gw (Demolab)	lan (Demolab)	Delete

Save

Cancel

Click **Save** to save the template settings.

## 4.6. Creating CPE templates for SD-WAN gateways

The SD-WAN gateway template includes parameters that are applied during registration or reboot of the device.

### 4.6.1. Create template for the vGW-11.

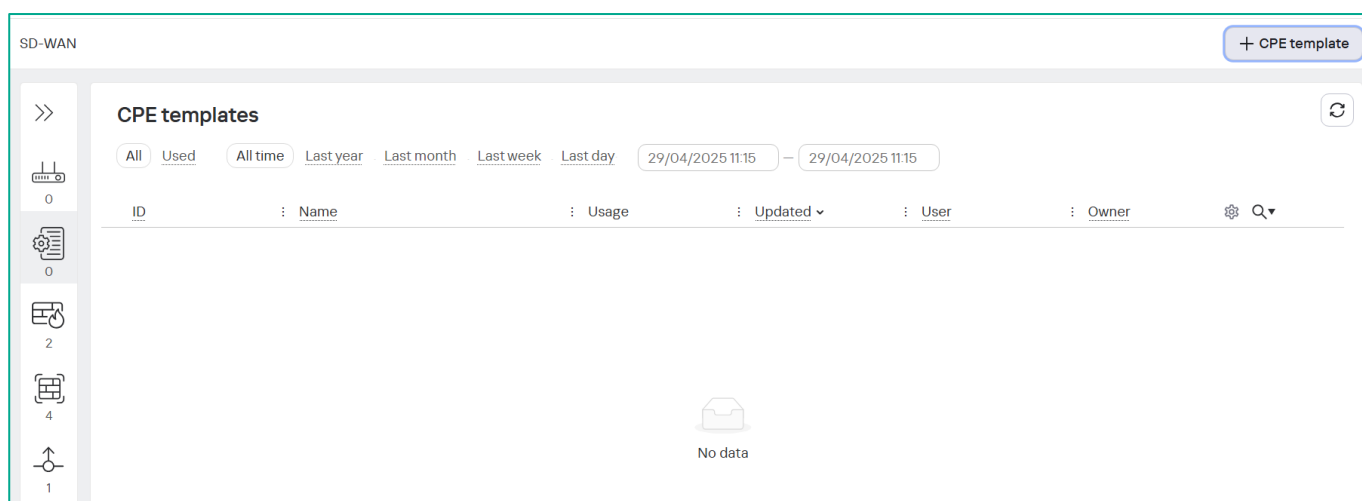
In this PoC separate CPE templates are used for each SD-WAN gateway.

Connect to the tenant self-service portal (PoC uses the tenant **Demolab**), to do this, click **Connect as Tenant** from the **Tenants** menu or connect to the SD-WAN orchestrator by the administrator of the created tenant.

**Note:** When CPE templates are created by an administrator from the administrator portal, they will not be available to users with tenant roles.

Go to **SD-WAN → CPE templates**.

Click **+ CPE Template**.



Specify vGW-11 template parameters:

- **Name:** vGW-11.
- **Type:** CPE.

Click **Create**.

New CPE template

×

Name

vGW-11

Type

CPE

▼

Create

Cancel

A separate template will be created for the vGW-12 gateway.

## 4.6.2. Set multipathing parameters in the vGW-11 template.

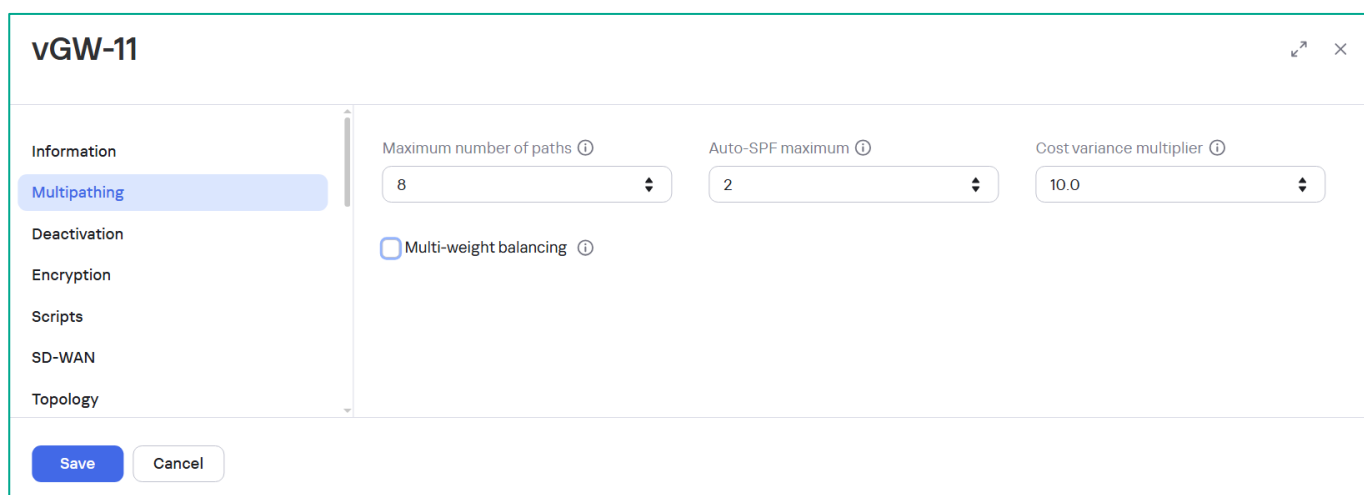
Open vGW-11 template.

Switch to **Multipathing** tab.

- Use default values: **8/2/10**.
- Disable **Multi Weight balancing** (paths will be added to the segments regardless of their cost) .

For more information, please refer to Kaspersky SD-WAN Online Help:

<https://support.kaspersky.com/help/SD-WAN/2.4/en-US/243185.htm>

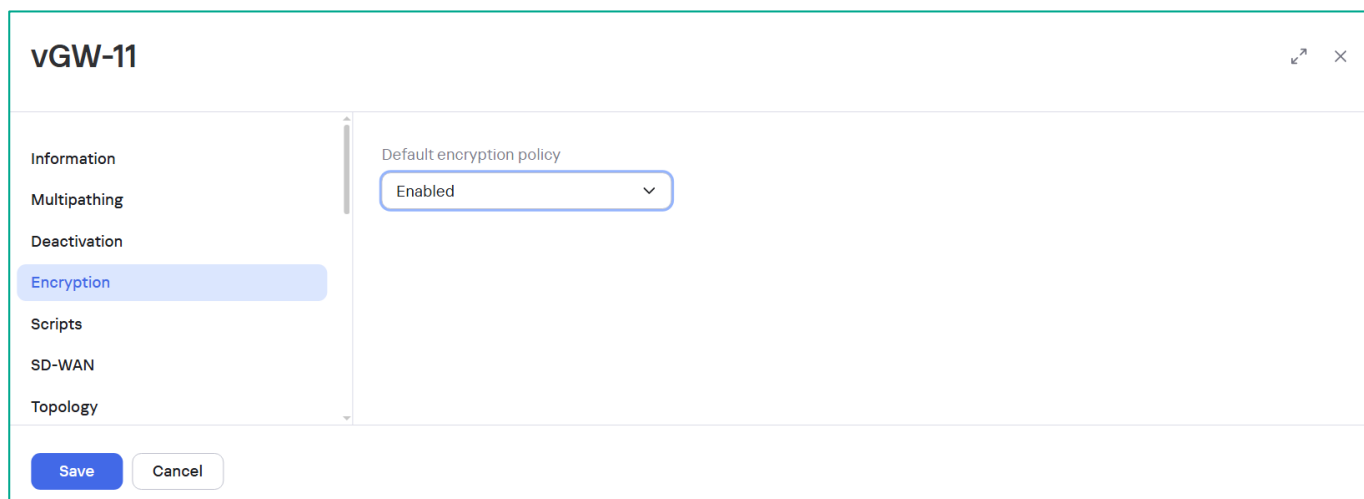


The screenshot shows the 'vGW-11' configuration window with the 'Multipathing' tab selected. The left sidebar lists 'Information', 'Multipathing', 'Deactivation', 'Encryption', 'Scripts', 'SD-WAN', and 'Topology'. The main area contains three input fields: 'Maximum number of paths' (set to 8), 'Auto-SPF maximum' (set to 2), and 'Cost variance multiplier' (set to 10.0). Below these is a checkbox for 'Multi-weight balancing' which is unchecked. At the bottom are 'Save' and 'Cancel' buttons.

## 4.6.3. Set encryption policy in the vGW-11 template.

Switch to the **Encryption** tab.

Set the encryption policy to **Enabled**.



The screenshot shows the 'vGW-11' configuration window with the 'Encryption' tab selected. The left sidebar lists 'Information', 'Multipathing', 'Deactivation', 'Encryption', 'Scripts', 'SD-WAN', and 'Topology'. The main area contains a 'Default encryption policy' dropdown menu set to 'Enabled'. At the bottom are 'Save' and 'Cancel' buttons.

4.6.4. Set SD-WAN parameters in the vGW-11 template.

Switch to **SD-WAN → General settings** tab.

Set the IP address to connect to the orchestrator (in the PoC it is orc1 public address, it is also possible to use a domain name to connect):

- **Orchestrator IP address /FQDN: 10.50.1.14.**
- **Orchestrator port: 443.**
- **OpenFlow transport: ssl.**
- **Control SD-WAN interface: sdwan0.**
- Change IP-address **192.168.7.1** in **Configuration URL** to **10.1.3.11**

## vGW-11

- Information
- Multipathing
- Deactivation
- Encryption
- Scripts
- SD-WAN**
- Topology
- Network
- DHCP
- BGP
- VRF
- OSPF
- Routing filters
- PBR
- BFD
- Static routes
- Multicast
- VRRP
- CFM
- Monitoring
- Transport services
- Log files

General settings Interfaces

### Connection to orchestrator

Orchestrator IP address/FQDN

Orchestrator port

☐ Backup orchestrator IP address and port

Orchestrator protocol

Update interval (sec)

Interactive update interval (sec)

Interactive mode timeout (sec)

### Connection to controller

OpenFlow transport

Control SD-WAN interface

☐ Preemption

Auto-reboot

Reboot timeout (sec)

Configuration URL

Save

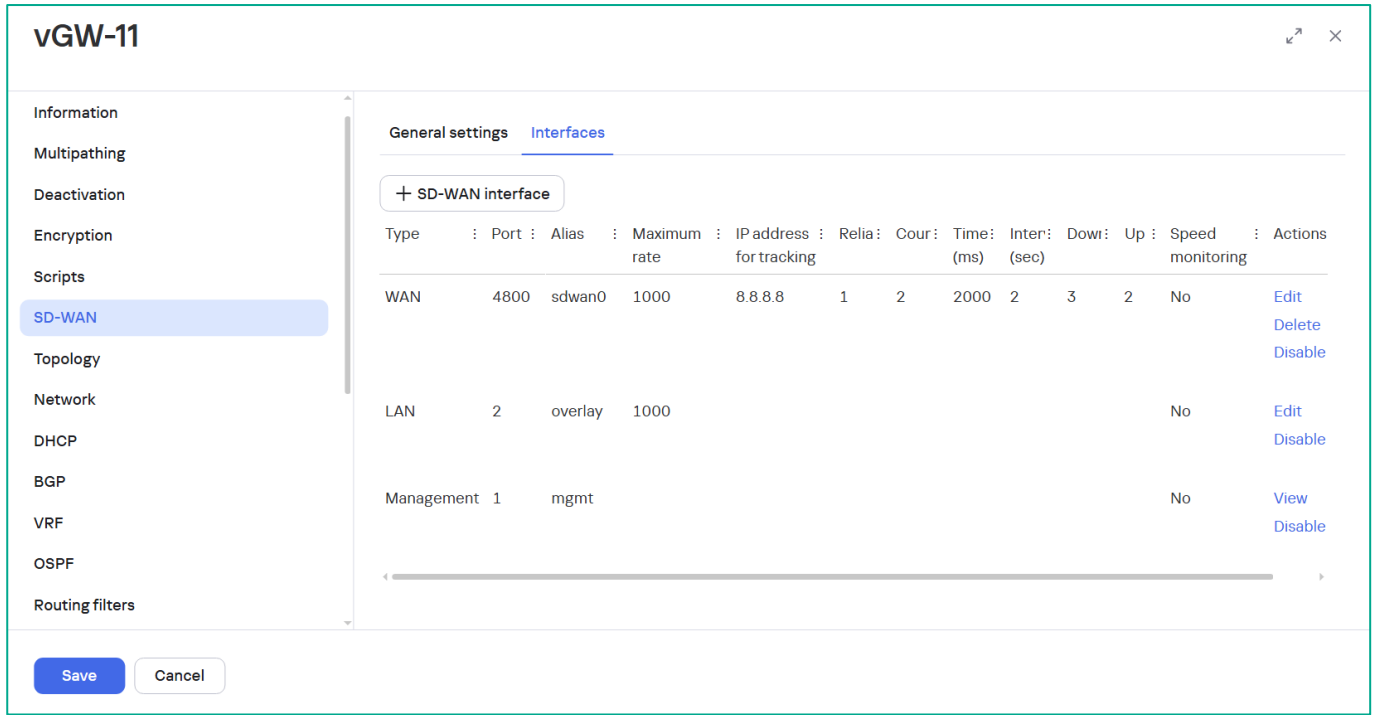
Cancel

#### 4.6.5. Configure SD-WAN interfaces in the vGW-11 template.

Switch to **SD-WAN → Interfaces** tab.

In this PoC, the SD-WAN gateways has only one external network interface.

Remove the **sdwan1** network interface (click **Delete** for the sdwan1 interface).



**vGW-11**

Information  
Multipathing  
Deactivation  
Encryption  
Scripts  
**SD-WAN**  
Topology  
Network  
DHCP  
BGP  
VRF  
OSPF  
Routing filters

General settings **Interfaces**

+ SD-WAN interface

Type	Port	Alias	Maximum rate	IP address for tracking	Relia	Cour	Time (ms)	Inter (sec)	Down	Up	Speed monitoring	Actions
WAN	4800	sdwan0	1000	8.8.8.8	1	2	2000	2	3	2	No	Edit Delete Disable
LAN	2	overlay	1000								No	Edit Disable
Management	1	mgmt									No	View Disable

Save Cancel

Set **sdwan0** network interface parameters.

Click **Edit** for the **sdwan0** interface.

Set the **IP** address for interface **tracking**. The default gateway IP address or **8.8.8.8** can be used.

Click **Save**.

If no tracking IP is accessible, the SD-WAN gateway or CPE devices will consider the network interface to be down and will not create tunnels through it. In this case, the route metric for this interface will be 21.



### SD-WAN interface ×

General settings QoS NAT and disjoint WAN underlay Controllers

Type  
WAN

OpenFlow port: 4800

Interface (alias): sdwan0

Maximum rate: 1000

IP address for tracking: 8.8.8.8 ×

IP address for fragmentation check: 1.1.1.1

+ Add

Reliability: 1

Interval (sec): 2

Count: 2

Timeout (ms): 2000

Down: 3

Up: 2

Speed monitoring: No

Save Cancel

4.6.6. Set CPE topology role in the vGW-11 template.

Switch to the **Topology** tab.

Set **Role** to **Gateway**.

### vGW-11 ↗ ×

Information

Multipathing

Deactivation

Encryption

Scripts

SD-WAN

**Topology**

Network

Role: Gateway

Save Cancel

#### 4.6.7. Configure network interfaces in the vGW-11 template.

Switch to the **Network** tab.

Add the following network interfaces (click **+ Network interface**):

- **sdwan0: eth0.**
- **lan: eth1.**
- **overlay: overlay.**
- **nfvmgmt: mgmt.**

Below is a description of the interface parameters.

vGW-11

Information

Multipathing

Deactivation

Encryption

Scripts

SD-WAN

Topology

Network

DHCP

BGP

VRF

OSPF

Routing filters

PBR

BFD

Static routes

+ Network interface

Alias	Zone	Interface name	Protocol	IP address/mask	MTU	Enable automatically	Actions
lan	lan	eth1	Static IPv4 address	IP address: 10.1.3.11 Mask: 255.255.255.0		Yes	Edit Delete Disable
nfvmgmt	mgmt_gw	mgmt	None			Yes	Edit Delete Disable
overlay	lan	overlay	Static IPv4 address	IP address: 172.16.1.11 Mask: 255.255.255.0		Yes	Edit Delete Disable
sdwan0	wan	eth0	Static IPv4 address	IP address: 10.1.4.11 Mask: 255.255.255.0		Yes	Edit Delete Disable

Save

Cancel

Add the **lan** network interface with following parameters:

- **Alias:** lan.
- **Zone:** lan.
- **Interface name:** eth1.
- **Protocol:** Static IPv4 address.
- **IPv4 address:** 10.1.3.11/24.

Click **Create**.

The screenshot shows the 'New network interface' configuration window. The 'Alias' is 'lan', the 'Zone' is 'lan (Demolab)', and the 'Interface name' is 'eth1'. The 'Protocol' is set to 'Static IPv4 address'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'IPv4 address and subnet mask input type' is set to 'Manually'. The 'IPv4 address' is '10.1.3.11' and the 'IPv4 netmask' is '255.255.255.0'. The 'Create' button is highlighted in blue.

Add the **overlay** network interface with following parameters:

- **Alias:** overlay.
- **Zone:** lan.
- **Interface name:** overlay.
- **Protocol:** Static IPv4 address.
- **IPv4 address:** 172.16.1.11/24.
- Check **Generate MAC address automatically**. With this setting, the interface's MAC address is automatically generated from the pool and saved after the device is rebooted, which eliminates the need to learn MAC addresses from neighboring devices and speeds up the convergence time of routing protocols.

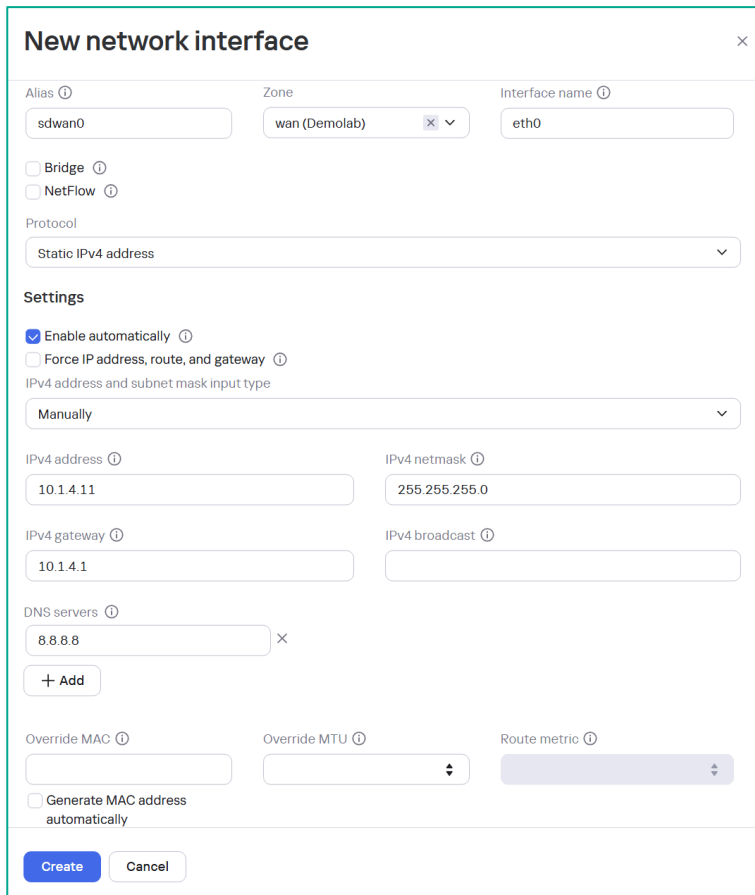
Click **Create**.

The screenshot shows the 'New network interface' configuration window for the 'overlay' interface. The 'Alias' is 'overlay', the 'Zone' is 'lan (Demolab)', and the 'Interface name' is 'overlay'. The 'Protocol' is set to 'Static IPv4 address'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'IPv4 address and subnet mask input type' is set to 'Manually'. The 'IPv4 address' is '172.16.1.11' and the 'IPv4 netmask' is '255.255.255.0'. The 'IPv4 gateway' and 'IPv4 broadcast' fields are empty. The 'DNS servers' section has an '+ Add' button. The 'Override MAC' field is empty, and the 'Generate MAC address automatically' checkbox is checked. The 'Override MTU' and 'Route metric' fields are empty. The 'Create' button is highlighted in blue.

Add the **sdwan0** network interface with following parameters:

- **Alias:** sdwan0.
- **Zone:** wan.
- **Interface name:** eth0.
- **Protocol:** Static IPv4 address.
- **IPv4 address:** 10.1.4.11/24.
- **IPv4 gateway:** 10.1.4.1.
- **DNS servers:** 8.8.8.8.

Click **Create**.



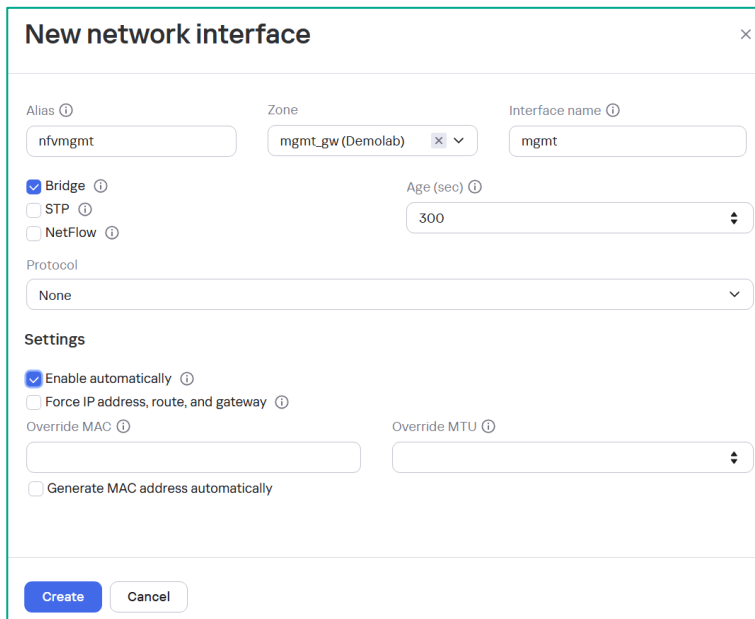
The screenshot shows the 'New network interface' configuration window. The 'Alias' field is set to 'sdwan0', the 'Zone' is 'wan (Demolab)', and the 'Interface name' is 'eth0'. The 'Bridge' checkbox is unchecked, and 'NetFlow' is also unchecked. The 'Protocol' is set to 'Static IPv4 address'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'IPv4 address and subnet mask input type' is set to 'Manually'. The 'IPv4 address' is '10.1.4.11' and the 'IPv4 netmask' is '255.255.255.0'. The 'IPv4 gateway' is '10.1.4.1' and the 'IPv4 broadcast' is empty. The 'DNS servers' field contains '8.8.8.8'. The 'Override MAC' field is empty, and 'Generate MAC address automatically' is unchecked. The 'Override MTU' field is empty, and the 'Route metric' is set to 1. The 'Create' button is highlighted in blue.

Add the **nfvmgmt** network interface with following parameters:

- **Alias:** nfvmgmt.
- **Zone:** mgmt\_gw.
- **Interface name:** mgmt.
- Check **Bridge**.
- **Age:** 300.
- **Protocol:** None.
- Check **Enable automatically**.

This interface is designed to provide connectivity between the mgmt network on the CPE and the orc1 host (required for SSH console operation from the orchestrator web interface).

Click **Create**.



The screenshot shows the 'New network interface' configuration window. The 'Alias' field is set to 'nfvmgmt', the 'Zone' is 'mgmt\_gw (Demolab)', and the 'Interface name' is 'mgmt'. The 'Bridge' checkbox is checked, and 'STP' and 'NetFlow' are unchecked. The 'Age (sec)' is set to 300. The 'Protocol' is set to 'None'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'Override MAC' field is empty, and 'Generate MAC address automatically' is unchecked. The 'Override MTU' field is empty. The 'Create' button is highlighted in blue.

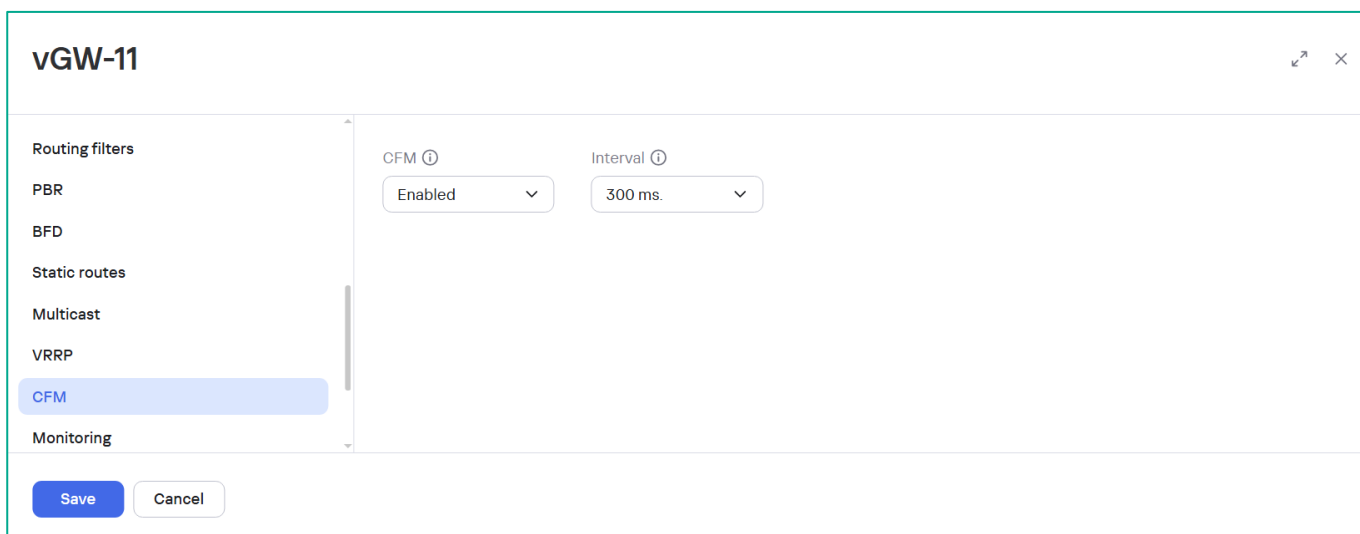
#### 4.6.8. Configure CFM in the vGW-11 template.

The Connectivity Fault Management functionality allows detecting unavailable links between CPE devices. When CFM is enabled, the CPE device sends Continuity Check Message control packets over its links at the specified intervals, and listens for response control packets on opposite-direction links. If response control packets do not arrive, the CPE device considers the link unavailable and starts transmitting traffic over a randomly selected available link.

Switch to **CFM** tab.

Set the following parameters:

- **CFM: Enabled** (enable CFM for all links).
- **Interval: 300 ms** (interval for sending control packets).



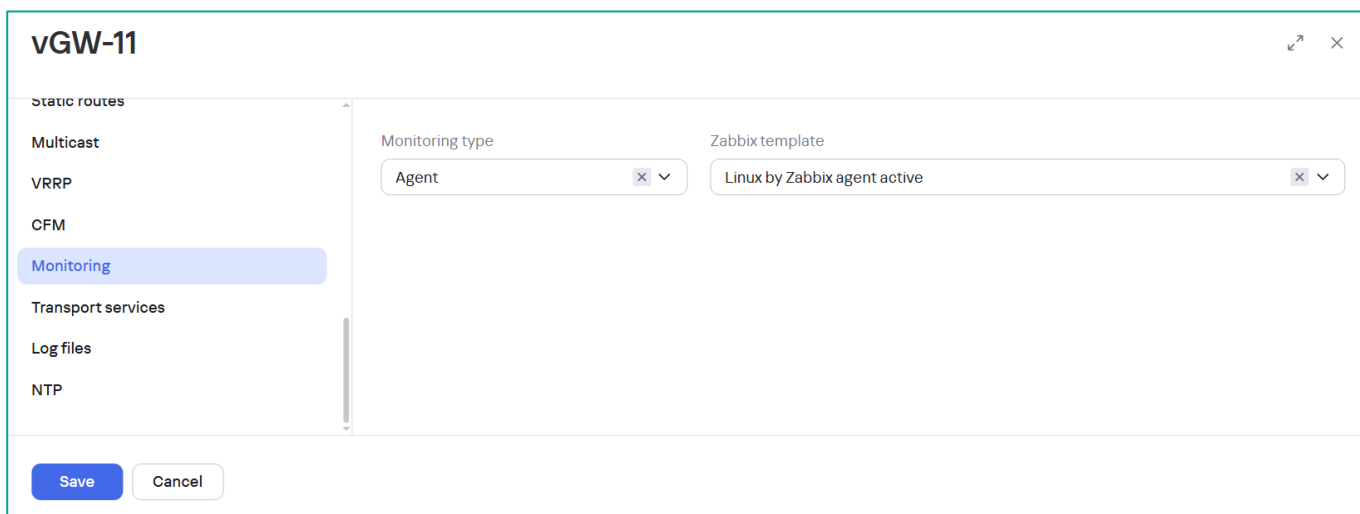
The screenshot shows the 'vGW-11' configuration window. On the left, a sidebar lists various configuration tabs: Routing filters, PBR, BFD, Static routes, Multicast, VRRP, CFM (highlighted in blue), and Monitoring. The main area displays the 'CFM' configuration. It features two dropdown menus: 'CFM' set to 'Enabled' and 'Interval' set to '300 ms.'. At the bottom, there are 'Save' and 'Cancel' buttons.

#### 4.6.9. Configure monitoring parameters in the vGW-11 template.

Switch to **Monitoring** tab.

Set the following parameters:

- **Monitoring type: Agent.**
- **Zabbix template: Linux by Zabbix agent active.**

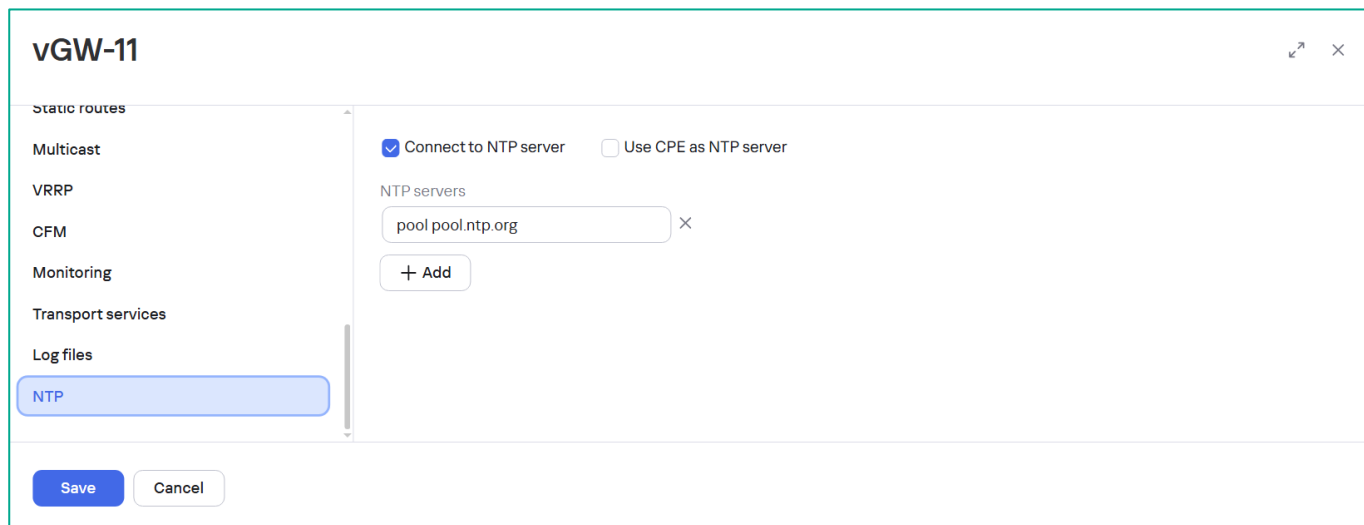


The screenshot shows the 'vGW-11' configuration window with the 'Monitoring' tab selected in the sidebar. The main area displays two dropdown menus: 'Monitoring type' set to 'Agent' and 'Zabbix template' set to 'Linux by Zabbix agent active'. At the bottom, there are 'Save' and 'Cancel' buttons.

4.6.10. Configure NTP in the vGW-11 template.

Switch to the **NTP** tab.

By default, the NTP client is enabled and **pool.ntp.org** is used.



The screenshot shows the 'vGW-11' configuration window. On the left is a sidebar with a list of configuration categories: Static routes, Multicast, VRRP, CFM, Monitoring, Transport services, Log files, and NTP. The 'NTP' category is selected and highlighted in blue. The main area of the window displays the NTP configuration. At the top, there are two checkboxes: 'Connect to NTP server' (which is checked) and 'Use CPE as NTP server' (which is unchecked). Below these is a section labeled 'NTP servers' containing a text input field with the value 'pool.pool.ntp.org' and a small 'x' icon to its right. Below the input field is a button labeled '+ Add'. At the bottom of the window, there are two buttons: 'Save' and 'Cancel'.

4.6.11. Create Prefix List in the vGW-11 template.

Switch to the **Routing filters** → **Prefix lists** tab.

Click **+ Prefix List**.

Set **Name**: **dc-net-list**.

Click **+ Rule**.

Add the following networks:

- **Seq 10 10.0.1.0/24.**
- **Seq 20 10.1.1.0/24.**
- **Seq 30 10.1.3.0/24.**
- **Seq 40 10.11.13.0/24.**

**Note:** If the mgmt subnet is changed in step 4.1.5, it is necessary to change the subnet to the new one in sequence 40.

## New prefix list

Name ⓘ

dc-net-list

+ Rule

Sequence	Network	Action	Greater or equal	Less or equal	
10	IP addre... ▾	10.0.1.0/24	Permit ▾		✕
20	IP addre... ▾	10.1.1.0/24	Permit ▾		✕
30	IP addre... ▾	10.1.3.0/24	Permit ▾		✕
40	IP addre... ▾	10.11.13.0/24	Permit ▾		✕

Create

Cancel

Click **Create**.

4.6.12. Create route map in the vGW-11 template.

Switch to the **Routing filters** → **Route maps** tab.

Click **+ Route Map**.

Set **Name**: **dc-route-map**.

Click **+ Rule** and set rule parameters.

- **Sequence**: **10**.
- **Action**: **Permit**.
- **Match Type**: **Prefix-list**.
- **Prefix list**: **dc-net-list**.

Click **Create**.

The screenshot shows the 'vGW-11' configuration page on the left and a 'New route map' dialog box on the right. The dialog box has a close button (X) in the top right corner. Inside the dialog, the 'Name' field is set to 'dc-route-map'. Below this is a '+ Rule' button. A table of rules is displayed with the following columns: Sequence, Action, Match type, Value, Change attribute, and New value. The first rule has Sequence '10', Action 'Permit', Match type 'Prefix-L...', Value '805dc020-24df-11', Change attribute 'None', and New value (empty). Below the table, there is a 'Prefix list' field set to 'dc-net-list' with a dropdown arrow and a close button (X). At the bottom of the dialog are 'Create' and 'Cancel' buttons. The background shows the 'vGW-11' configuration page with a sidebar containing 'VRF', 'OSPF', 'Routing filters' (selected), 'PBR', 'BFD', 'Static routes', 'Multicast', 'VRRP', 'CFM', and 'Monitoring'. The 'Routing filters' section is expanded, showing 'Access control' and '+ Route' buttons.

Sequence	Action	Match type	Value	Change attribute	New value
10	Permit	Prefix-L...	805dc020-24df-11	None	



4.6.13. Configure BGP in the vGW-11 template.

Switch to the **BGP** menu.

Set BGP **Autonomous System** → **65500**.

Click **+ BGP** to add a new BGP instance.

vGW-11

Deactivation

Encryption

Scripts

SD-WAN

Topology

Network

DHCP

BGP

VRF

OSPF

Autonomous System

65500

Default BGP Instance with VRF ⓘ

+ BGP

State	VRF	Router ID	BGP neighbor	Peer groups	Route Distinguisher	Export routes	Import routes	A
No data								

SaveCancel

Switch to **General settings** tab.

- Set the BGP parameters:
- **BGP: Enabled.**
- **Router ID: 172.16.1.11** (overlay interface IP address).
- **Maximum Paths: 2.**
- **Graceful Restart.**
- **Default IPv4 Unicast.**
- **BGP Timers:**
  - **Keepalive: 10.**
  - **Holdtime: 30.**

Apply **Route Map: dc-route-map** for **Connected** routes redistribution.

## BGP instance

[General settings](#)
[Neighbors](#)
[Peer groups](#)
[Route leaking](#)

BGP

Enabled

VRF

main/254

AS

65500

Router ID

172.16.1.11

☐ Router ID from IP pool

Maximum paths

2

☐ Always compare MED
 ☒ Graceful restart (helper mode)
 ☒ Use default IPv4 unicast routes

☒ BGP timers

Keepalive (sec)

10

Holdtime (sec)

30

Route redistribution

☐ Kernel
 

Route map

Metric

☒ Connected
 

Route map

dc-route-map

Metric

Save

Cancel

4.6.14. Configure the BGP peer group to establish BGP neighborhood with CPE devices.

Switch to the **BGP → Peer groups** tab.

Click **+ Peer group**.

Set the peer group parameters:

- **Name:** CPE.
- **BGP Range:** 172.16.1.0/24 (overlay interface subnet).
- **Remote AS:** 65500.

The screenshot shows the 'New peer group' dialog box with the 'General settings' tab selected. The 'Name' field contains 'CPE'. There is an unchecked checkbox for 'Shutdown peer group'. The 'BGP range' field contains '172.16.1.0/24'. The 'Remote AS' field contains '65500'.

Switch to the **Advanced Settings** tab.

Enable **Route Reflector Client**.

The screenshot shows the 'New peer group' dialog box with the 'Advanced settings' tab selected. Under the 'General settings' section, the 'Route reflector client' checkbox is checked. Other checkboxes include 'Soft-reconfiguration inbound', 'Allow AS in', 'Next-hop self', 'Attribute unchanged AS path', 'Attribute unchanged next-hop', and 'Attribute unchanged MED'. Under the 'Advanced settings' section, there are three dropdown menus for 'Local AS', 'Weight', and 'Maximum prefix'. At the bottom, there are checkboxes for 'Send community' and 'Default originate', and two buttons: 'Create' and 'Cancel'.

Click **Create**.

4.6.15. Create BGP neighborships to R13 and vGW-12 from vGW-11

Switch to the **Neighbors** tab and click **+ BGP Neighbor**.

Create 2 BGP neighbors.

Set the BGP neighbor parameters:

- **Name: R13.**
- **Neighbor IP: 10.1.3.13.**
- **Remote AS: 65613.**
- **Name: vGW-12.**
- **Neighbor IP: 10.1.3.12.**
- **Remote AS: 65500.**

## BGP instance

General settings
Neighbors
Peer groups
Route leaking

+ BGP neighbor

Neighbor IP	Name	Description	Remote AS	Shutdown	Weight	Actions
10.1.3.12	vGW-12		65500	No		<a href="#">Edit</a> <a href="#">Delete</a>
10.1.3.13	R13		65613	No		<a href="#">Edit</a> <a href="#">Delete</a>

Save
Cancel

Click **Save** to save BGP instance settings.

Set **Default BGP Instance with VRF: main/254** (Default VRF for BGP instances, used for backward compatibility with older CPE versions).

Click **Save** to save template settings.

## vGW-11

Deactivation
Encryption
Scripts
SD-WAN
Topology
Network
DHCP
BGP
VRF
OSPF
Routing filters

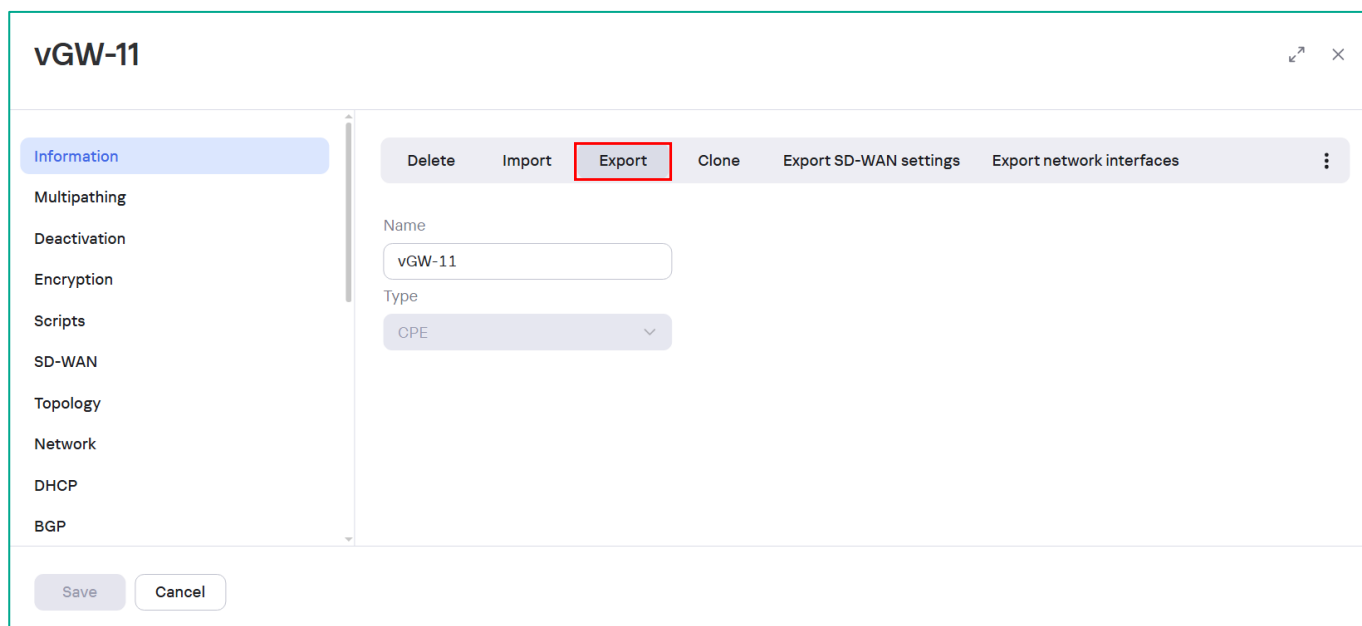
Autonomous System
65500
Default BGP Instance with VRF
main/254
+ BGP

	Router ID	BGP neighbor	Peer groups	Route Distinguisher	Export routes	Import routes	Actions
4	172.16.1.11	2	1	65500:254	Off	Off	<a href="#">Edit</a> <a href="#">Delete</a>

Save
Cancel

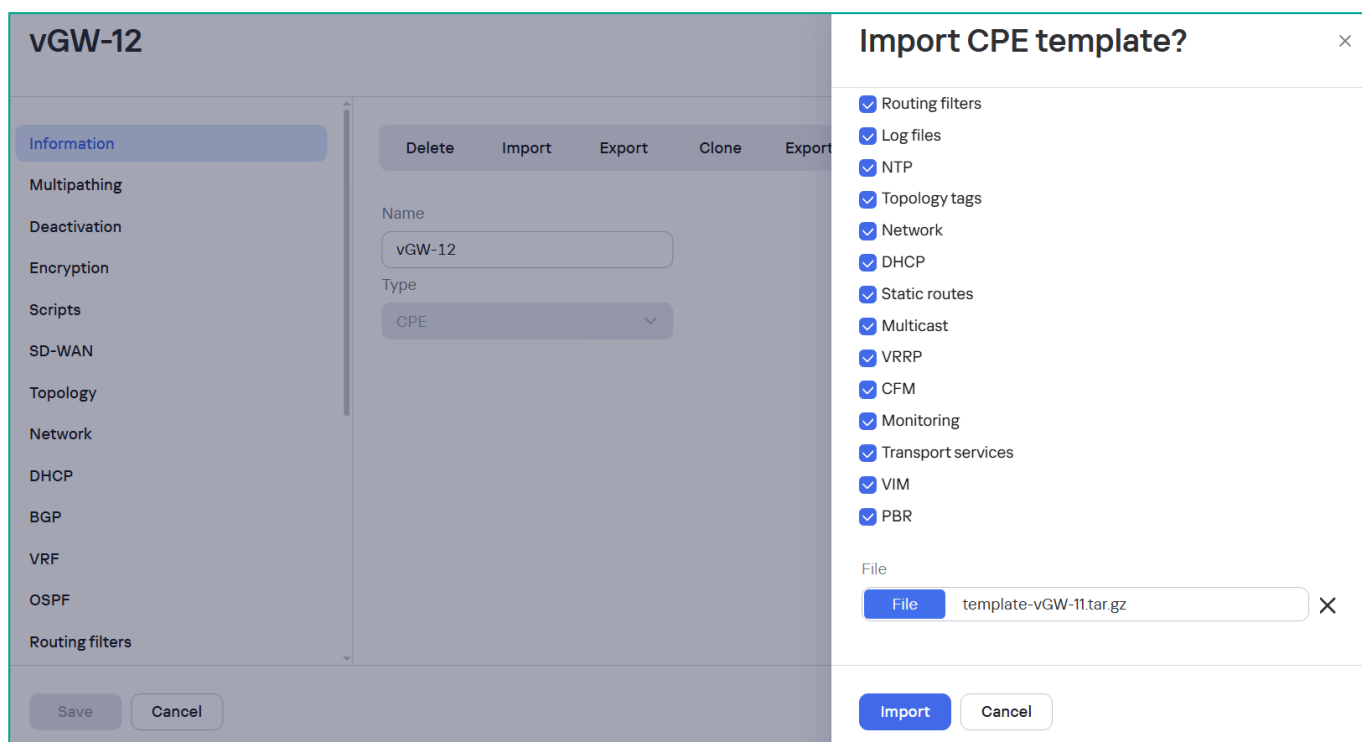
4.6.16. Export the vGW-11 SD-WAN gateway template.

Select **vGW-11** template, go to **Information** tab, click **Export** and download template archive.



4.6.17. Create a template for the vGW-12 gateway.

Repeat steps 4.6.1 – 4.6.15 with vGW-12 parameters or import the vGW-11 template. To do this create new vGW-12 template, then click **Import** and select: **template-vGW-11.tar.gz** file. Click **Import**.



Also, you can **Clone** vGW-11 template to the vGW-12 template.

4.6.18. Change the imported template for the vGW-12 gateway.

At the **SD-WAN → General settings** change IP address to **10.1.3.11** in **Configuration URL** to **10.1.3.12**.

**vGW-12**

Information  
Multipathing  
Deactivation  
Encryption  
Scripts  
**SD-WAN**  
Topology  
Network  
DHCP  
BGP  
VRF  
OSPF

**Connection to controller**

OpenFlow transport  
SSL

Control SD-WAN interface  
sdwan0

☐ Preemption

Auto-reboot  
No

Reboot timeout (sec)  
86400

Configuration URL  
http://10.1.3.12/cgi-bin/config?payload={config}

Save Cancel

In the **Network** menu update the interfaces settings:

- **sdwan0 IPv4 address: 10.1.5.12/24.**
- **sdwan0 IPv4 gateway: 10.1.5.1.**
- **lan IPv4 address: 10.1.3.12/24.**
- **overlay IPv4 address: 172.16.1.12/24.**

**vGW-12**

Information  
Multipathing  
Deactivation  
Encryption  
Scripts  
SD-WAN  
Topology  
**Network**  
DHCP  
BGP  
VRF  
OSPF  
Routing filters  
PBR  
BFD

+ Network interface

Alias	Zone	Interface name	Protocol	IP address/mask	MTU	Enable automaticall	Actions
lan	lan	eth1	Static IPv4 address	IP address: 10.1.3.12 Mask: 255.255.255.0		Yes	Edit Delete Disable
nfvmgmt	mgmt_gw	mgmt	None			Yes	Edit Delete Disable
overlay	lan	overlay	Static IPv4 address	IP address: 172.16.1.12 Mask: 255.255.255.0		Yes	Edit Delete Disable
sdwan0	wan	eth0	Static IPv4 address	IP address: 10.1.5.12 Mask: 255.255.255.0 GW: 10.1.5.1		Yes	Edit Delete Disable

Save Cancel

Open BGP instance for editing: go to **BGP** and click **Edit** for BGP instance.

**vGW-12**

Encryption

Scripts

SD-WAN

Topology

Network

DHCP

**BGP**

VRF

OSPF

Routing filters

Autonomous System: 65500

Default BGP Instance with VRF: main/254

+ BGP

State	VRF	Router ID	BGP neighbor	Peer groups	Route Distinguisher	Export routes	Import routes	Actions
Enabled	main/254	172.16.1.11	2	1	65500:254	Off	Off	<a href="#">Edit</a> <a href="#">Delete</a>

Save Cancel

In the BGP instance change **Router ID: 172.16.1.12**.

**BGP instance**

General settings Neighbors Peer groups Route leaking

BGP: Enabled

VRF: main/254

AS: 65500

Router ID: 172.16.1.12

☐ Router ID from IP pool

Maximum paths: 2

☐ Always compare MED ☐ Graceful restart (helper mode) ☒ Use default IPv4 unicast routes

Save Cancel

At the **Neighbors** tab update BGP neighborships configuration:  
Change **vGW-12** to **vGW-11** (**Neighbor IP: 10.1.3.11**).

BGP instance

General settings

Neighbors

Peer groups

Route leaking

+ BGP neighbor

Neighbor IP	Name	Description	Remote AS	Shutdown	Weight	Actions
10.1.3.11	vGW-11		65500	No		<a>Edit</a> <a>Delete</a>
10.1.3.13	R13		65613	No		<a>Edit</a> <a>Delete</a>

Save

Cancel

Click **Save** in the BGP instance, then click **Save** to save gateway template.



## 4.7. Importing CA certificates for CPE devices

To prevent MITM (man-in-the-middle) attacks, when communicating with the orchestrator, the CPE device checks whether the orchestrator certificate can be trusted. By default, root certificates of public certificate authorities are installed on devices. If your orchestrator is using a certificate signed by a public certificate authority, you do not need to install an additional certificate on the devices. Otherwise, you must add the public root certificate used by the orchestrator on the devices by uploading the certificate to the orchestrator web interface.

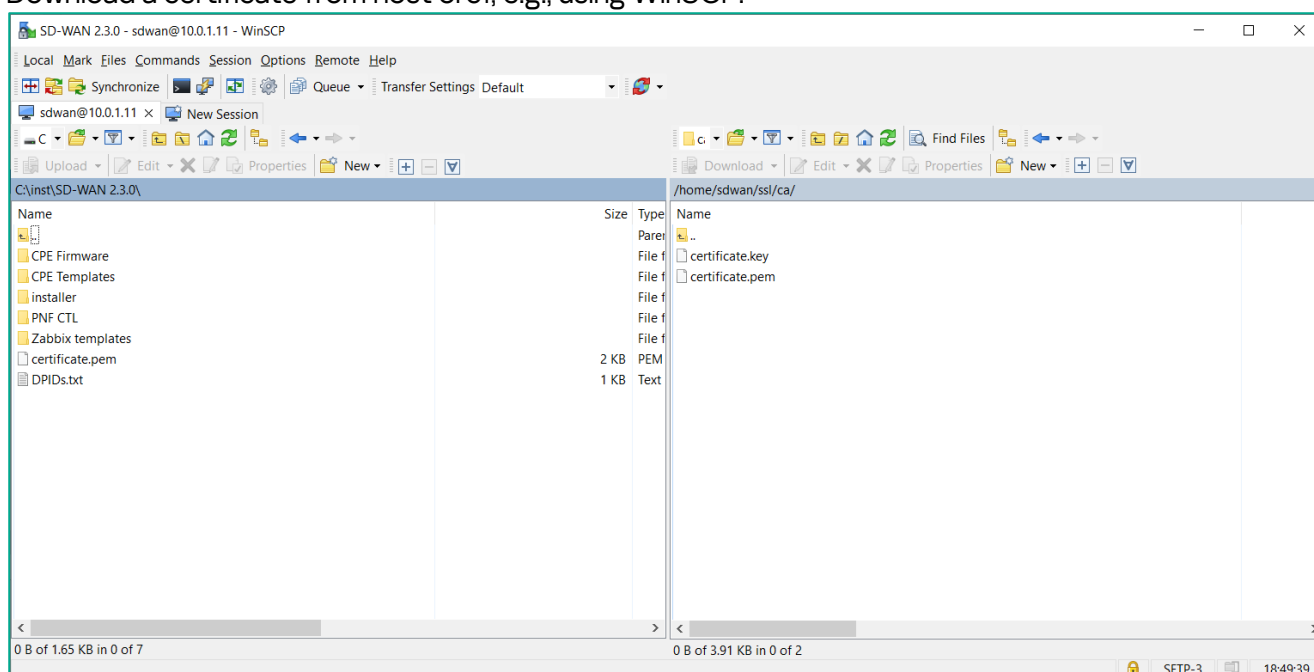
For more information, please refer to Kaspersky SD-WAN Online Help:

<https://support.kaspersky.com/help/SD-WAN/2.4/en-US/270629.htm>

### 4.7.1. Download root CA from orc1 host.

During the installation of the SD-WAN management system, the root CA certificate was saved to the file:  
**/home/sdwan/ssl/ca/certificate.pem.**

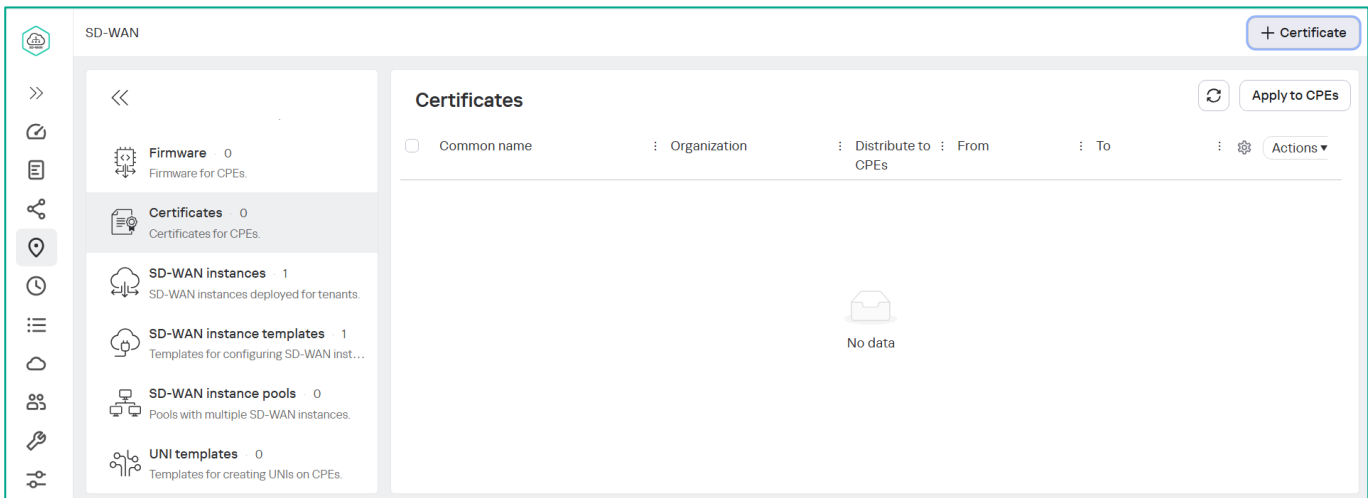
Download a certificate from host orc1, e.g., using WinSCP.



4.7.2. Upload root CA to the orchestrator.

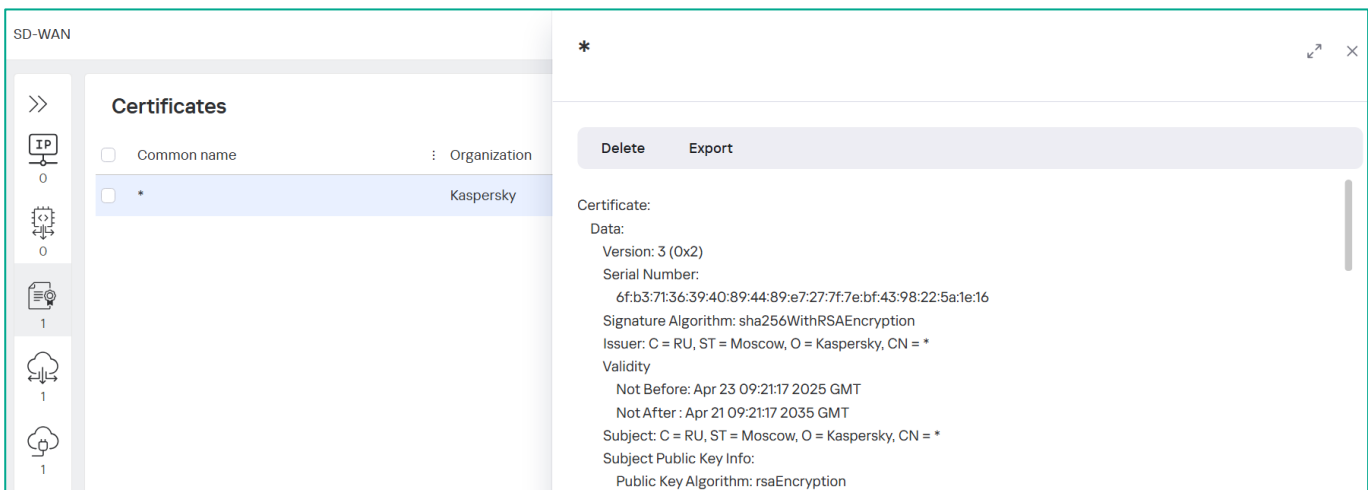
With administrator user go to the **SD-WAN → Certificates**.

Click **+ Certificate**, then select **.pem** certificate file for upload.



4.7.3. Verify loaded certificate.

Click the certificate to view it.



## 4.8. SD-WAN gateways initial configuration

4.8.1. Deploy VMs **vGW-11** and **vGW-12** from CPE vKESR-M2 image(knaas-cpe.<release\_name>.combined.amd64-vkesr-m2.vKESR-M2-esxi.tar.gz).

### Deploy OVF Template

- Select an OVF template**
- Select a name and folder
- Select a compute resource
- Review details
- Select storage
- Ready to complete

### Select an OVF template

Select an OVF template from remote URL or local file system

Enter a URL to download and install the OVF package from the Internet, or browse to a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

☐ URL

☒ Local file

4 files

Virtual machine resources for the vKESR-M2:

- 4 x CPU.
- 8 Gb RAM.

### Deploy OVF Template

- Select an OVF template
- Select a name and folder
- Select a compute resource
- Review details**
- Select storage
- Select networks
- Customize template
- Ready to complete

### Review details

Verify the template details.

The OVF package contains advanced configuration options, which might pose a security risk. Review the advanced configuration options below. Click next to accept the advanced configuration options.

Publisher	No certificate present
Product	VCPE
Vendor	Kaspersky Lab
Description	vKESR-M2 vCPU - 4 vRAM - 8192M knaas-cpe_2.24.09.release.23.combined.amd64-vkesr-m2
Download size	Unknown
Size on disk	Unknown (thin provisioned) 1.0 GB (thick provisioned)
Extra configuration	guestinfo.urlactivated = false nvram = ovf:/file/file2

Assign networks according to the PoC topology in Figure 2:

- **WAN1: DC-EDGE1.**
- **LAN1: DC-PERIM.**
- (Optionally) Delete WAN2.

Deploy OVF Template

- 1 Select an OVF template
- 2 Select a name and folder
- 3 Select a compute resource
- 4 Review details
- 5 Select storage
- 6 Select networks**
- 7 Customize template
- 8 Ready to complete

### Select networks

Select a destination network for each source network.

Source Network	Destination Network
LAN1	DC-PERIM
WAN1	DC-EDGE1
WAN2	Browse ...

3 items

#### IP Allocation Settings

IP allocation: Static - Manual

IP protocol: IPv4

CANCEL BACK NEXT

Repeat the previous steps and deploy the vGW-12 virtual machine from the vKESR-M2 image

## 4.8.2. Configure **lan** network interface

Open the console to the vGW-11 virtual machine.

The vi text editor is installed on the SD-WAN gateway:

- Press the **i** key to enter edit mode.
- Press the **Esc** key to return to command mode.
- **:wq** - to save the changes and exit.
- **:q!** - to exit without changes.

Edit the network configuration file:

```
vi /etc/config/network
```

It is necessary to configure the **lan** network interface to apply the configuration URL from the mgmt workstation.

Change the IP address of the vGW-11 **lan** interface according to Table 1 in section 2.2.

Change **ifname** in **lan** to **eth1**.

```
config interface 'ovs_lan'
    option device 'ovs-lan'
    option proto 'none'

config interface 'lan'
    option type 'bridge'
    option proto 'static'
    option ipaddr '10.1.3.11'
    option netmask '255.255.255.0'
    option ifname 'eth1'
    option auto '1'
    option force_link '1'

config interface 'sdwan0'
    option device 'eth0'
    option proto 'dhcp'
    option metric '100'
```

Restart network service:

```
/etc/init.d/network restart
```

Check the applied settings:

```
ip -br a
```

```
root@8000005056891685:/# ip -br a
lo                UNKNOWN      127.0.0.1/8  ::1/128
eth0              UP           10.1.3.176/24 fe80::250:56ff:fe89:1685/64
eth1              UP
ip6tnl0@NONE      DOWN
gre0@NONE         DOWN
gretap0@NONE      DOWN
erspan0@NONE      DOWN
ip6gre0@NONE      DOWN
bond0             DOWN
br-lan            UP           10.1.3.11/24 fe80::250:56ff:fe89:adaa/64
overlay@ovs-lan   UP           fe80::1494:d9ff:fe07:f0fb/64
ovs-lan@overlay   UP           fe80::78a2:65ff:fe39:5fcf/64
mgmt@ovs-mgmt     UP           fe80::bc5c:75ff:feca:e72f/64
ovs-mgmt@mgmt     UP           fe80::98f8:51ff:fe4d:5ba/64
```

Repeat previous step to configure **lan** interface for vGW-12.

After registration (applying the configuration URL in section 4.9) SD-WAN gateways will receive and replace the network settings according to the settings in step 4.6.7.

**Note:** You can also apply the configuration URL when deploying gateways from OVF. To obtain URL, you must first create a CPE using previously created gateway templates with a random DPID and copy the configuration URL (described in section 4.9). When configuring the CPE, you must use default IP address for the Configuration URL - 192.168.7.1. After booting, the CPE will appear in the orchestrator with the status Unknown. Next, open the CPE, click Register in the Configuration menu, and then configure the CPE settings as described in section 4.9.

## 4.9. SD-WAN gateways registration

4.9.1. Add vGW-11 device to the orchestrator.

Go to **SD-WAN** → **CPE**.

Click **+ CPE**.

Specify vGW-11 device parameters:

- CPE name: **vGW-11**.
- **DPID**: The device DPID is displayed in the command line of the CPE device.
- **Transport tenant**: **Demolab** (use tenant, created in section 4.4).
- **Customer tenant**: **Demolab** (use tenant, created in section 4.4).
- **CPE template**: **vGW-11** (gateway template, created in section 4.6).
- **Firewall template**: **gateway\_firewall\_template** (firewall template, created in section 4.5).

The CPE device connects to the controller of the SD-WAN instance that has been deployed for the transport tenant. The customer tenant can manage the CPE device in its self-service portal. Several customer tenants could be added to a single transport tenant.

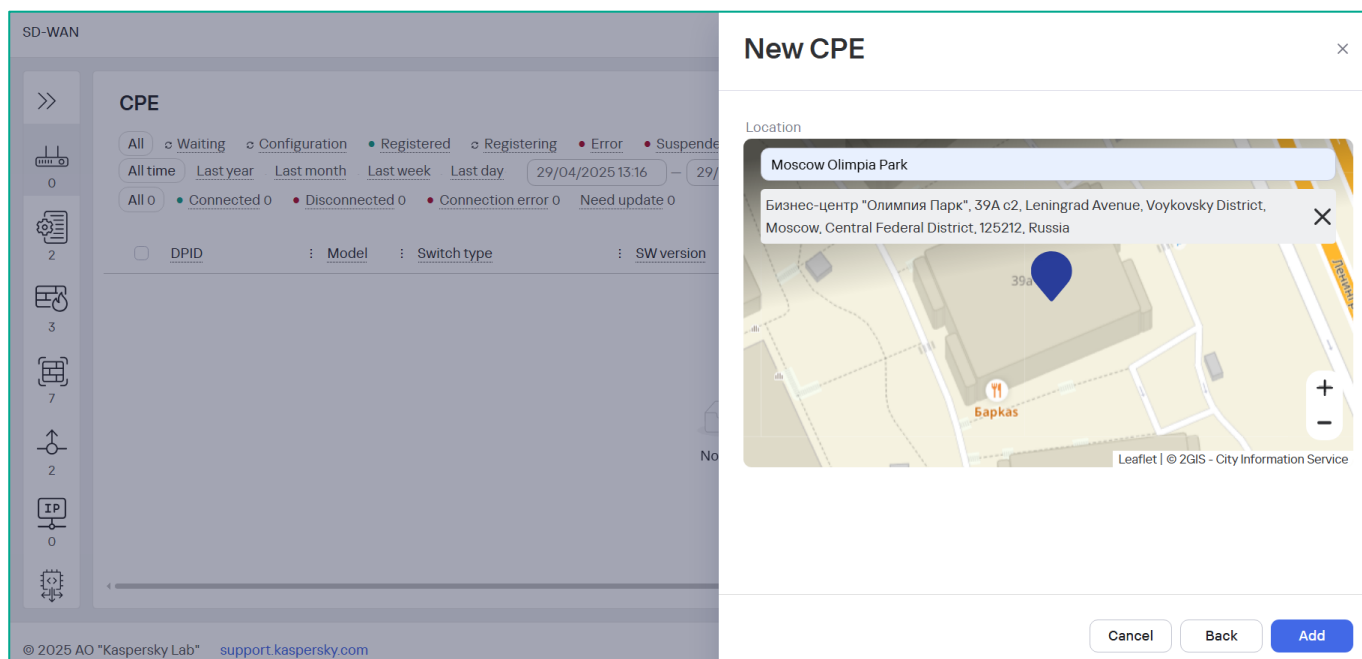
Click **Next**.

The screenshot displays the 'New CPE' configuration window in the Kaspersky SD-WAN interface. The window is titled 'New CPE' and features a close button (X) in the top right corner. The form contains the following fields and values:

- Name:** vGW-11
- DPID:** 8000005056AA9EA5
- State:** Enabled (dropdown menu)
- Description:** (empty text field)
- Transport tenant:** Demolab (dropdown menu)
- Customer tenant:** Demolab (dropdown menu with a clear button 'X')
- UNI template:** (empty dropdown menu)
- CPE template:** vGW-11 (dropdown menu)
- NetFlow template:** Default NetFlow template (dropdown menu)
- Firewall template:** gateway\_firewall\_template (Demolab) (Demolab) (dropdown menu)

At the bottom of the form, there are three buttons: 'Cancel', 'Back', and 'Next'. The background shows the 'SD-WAN' dashboard with a 'CPE' section and a list of devices.

Set the device location (optionally).

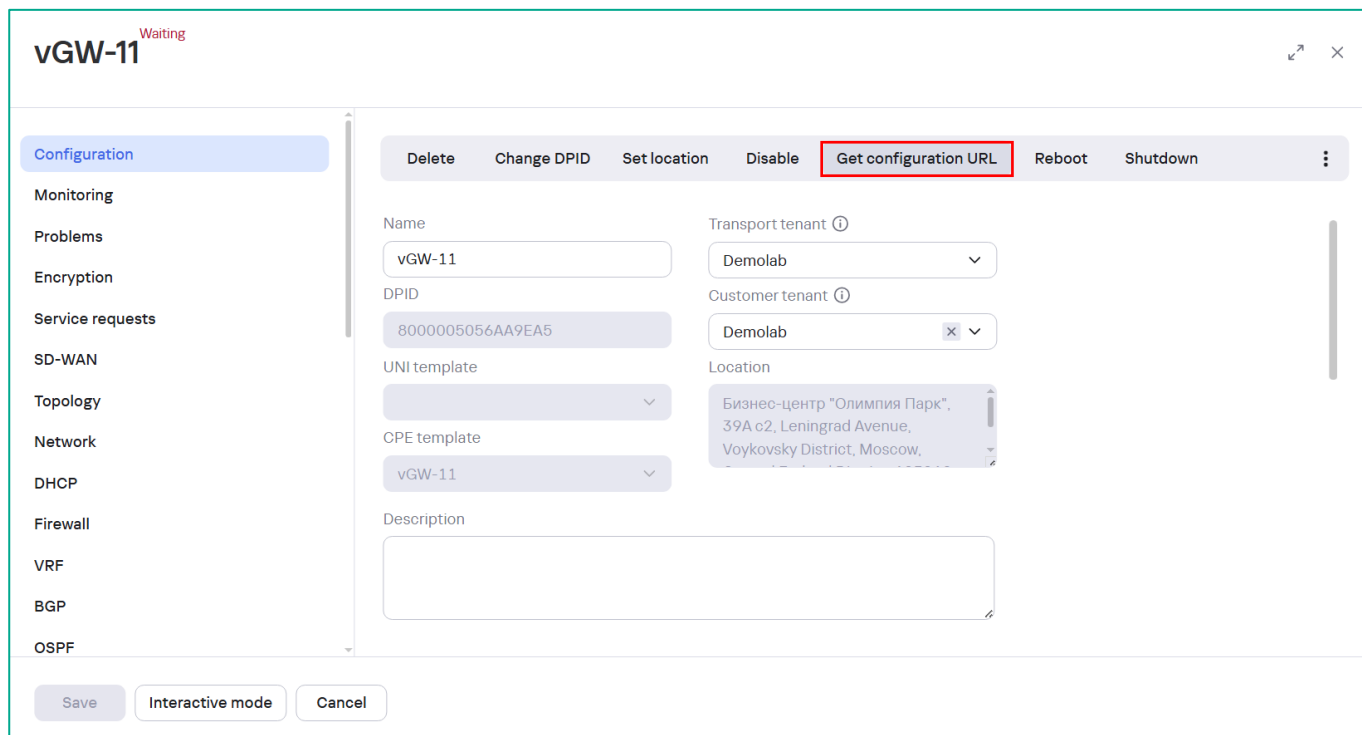


Click **Add**.

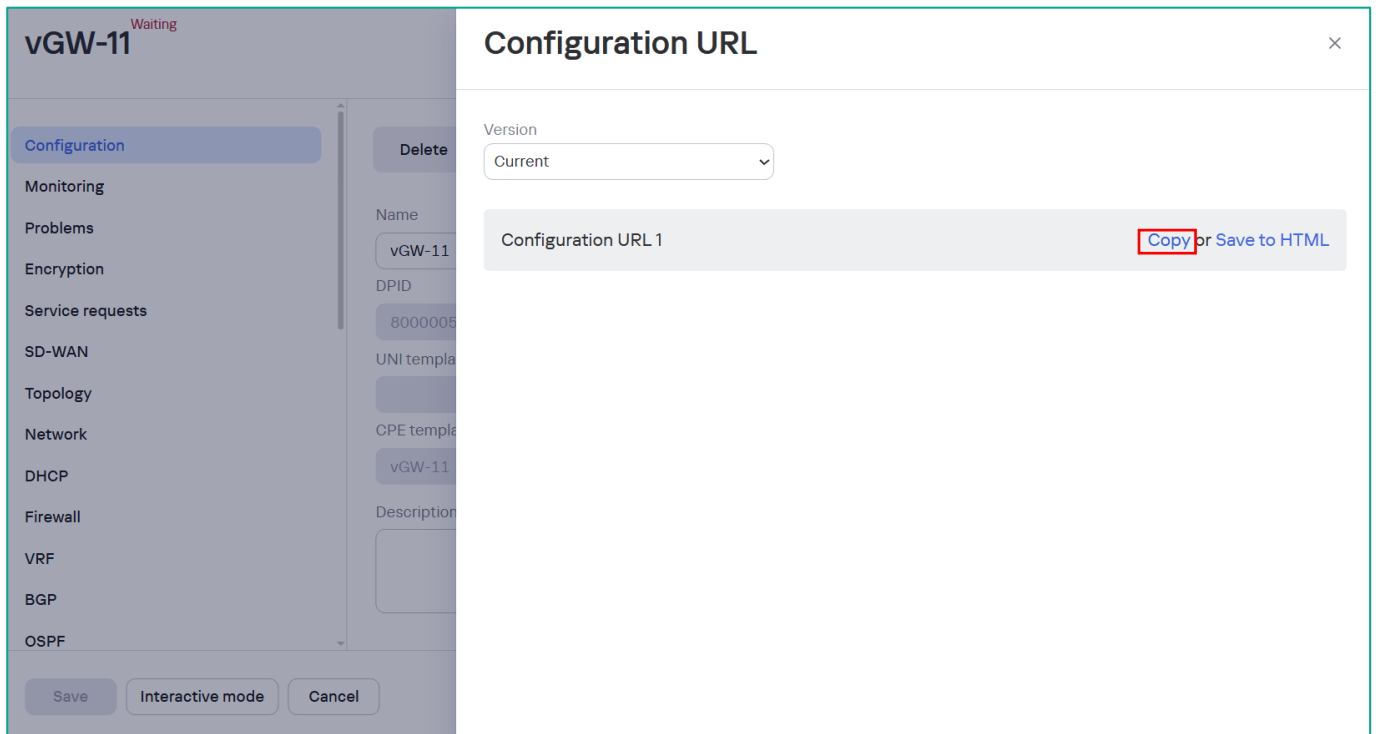
4.9.2. Configure vGW-11 gateway with Configuration URL.

Generate a Configuration URL for device registration.

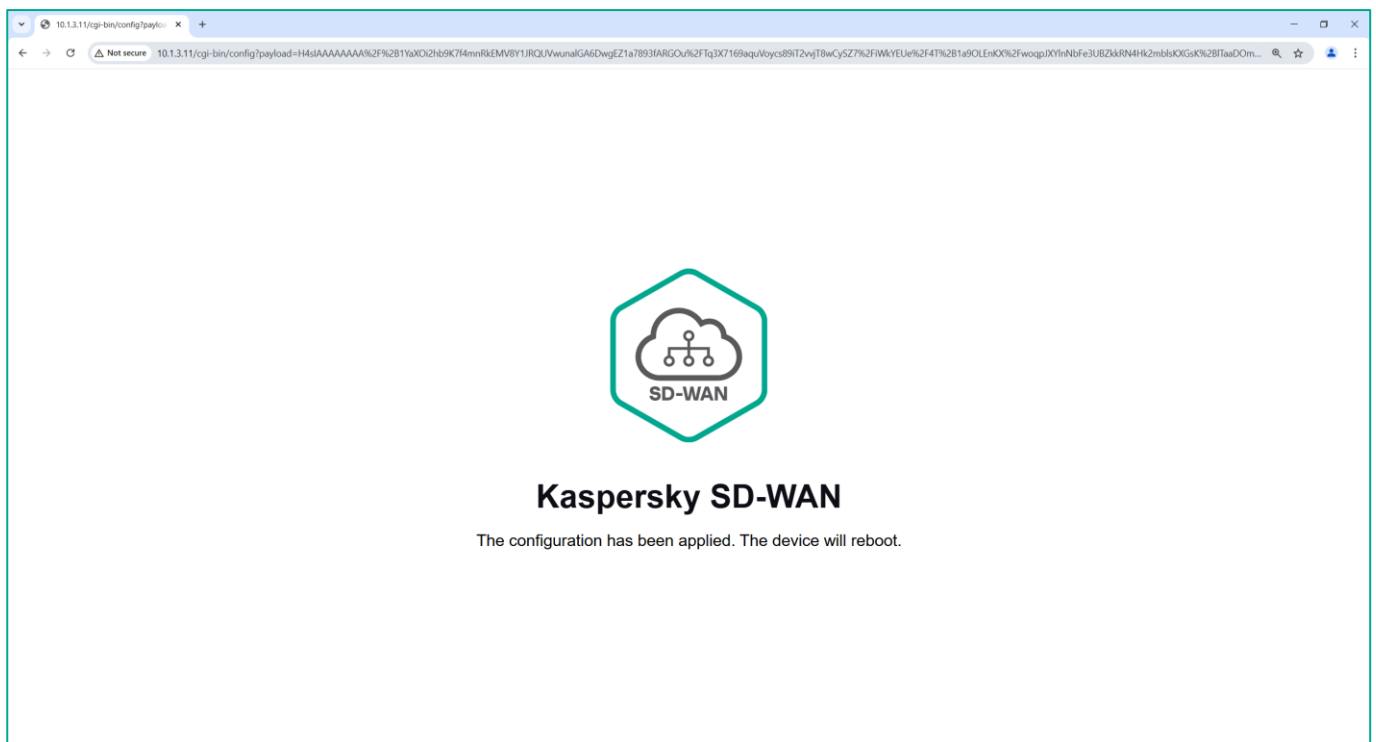
Go to **SD-WAN → CPE**, select vGW-11 and click **Get Configuration URL**.



Copy the URL.



Perform vGW-11 configuration. Open the copied link in browser's address bar (you must have network connectivity with the gateways lan interface) to apply settings to the vGW-11.





After reboot the CPE status will change to **Registering**.

**vGW-11** Registering

Configuration | Monitoring | Problems | Encryption | Service requests | SD-WAN | Topology | Network | DHCP | Firewall | VRF | BGP | OSPF

Delete Show password Get configuration URL Reboot Shutdown Export SD-WAN settings Export network interfaces

Name: vGW-11 Transport tenant: Demolab UNI template:   
 DPID: 8000005056AA9EA5 Customer tenant: Demolab CPE template: vGW-11   
 Location: Бизнес-центр "Олимпия Парк", 39А с2, Leningrad Avenue, Voykovsky District, Moscow,   
 Description:   
 NetFlow template: Default NetFlow template   
 Firewall template: gateway\_firewall\_template (De...

Save Interactive mode Cancel

The **Service Request** tab will appear. New service request will be created for CPE registration.

**vGW-11** Registered

Configuration | Monitoring | Problems | Encryption | Service requests

Reload service requests Cancel all service requests Delete all service requests

Name	Created	Task ID	Time	Status	Actions
CpeRegistration	29/04/2025 13:26:55	9711850a-1e48-4298-a00f-4b0c74fdd4e4	1m 31s	Executed	Delete

Click the **Task ID** for registration details.

CpeRegistration

Created: 29/04/2025 13:26:55

Task ID: 9711850a-1e48-4298-a00f-4b0c74fdd4e4

Time: 1m 31s

Status: Executed

Name	Status	Time	Attributes
CommutatorAttachCommand	Executed	1m 20s	cluster: SD-WAN Cluster [Demolab: 4f7461d3-0a4b-
CommutatorRenameCommand	Executed	0	name: vGW-11: 8000005056AA9EA5
CommutatorUpdatePortsStateSet	Executed	0	
CommutatorUpdatePortStateCommand	Executed	0	number: 4800
CommutatorSetLinksEncryptionCommand	Executed	0	encrypted: true
CommutatorSetCfmCommand	Executed	0	cfmEnabled: true
CommutatorUpdatePublicPortSettingsSet	Executed	0	
CommutatorUpdatePublicPortSettingsCommand	Executed	0	number: 4800
CommutatorSetGeoAddressCommand	Executed	0	

Refresh

Cancel

CPE device will change status to **Registered** and **Connected**.

CPE

Export to CSV...

All

Waiting

Configuration

Registered

Registering

Error

Suspended

Unknown

All time

Last year

Last month

Last week

Last day

29/04/2025 13:16

29/04/2025 13:16

All 1

Connected 0

Disconnected 0

Connection error 0

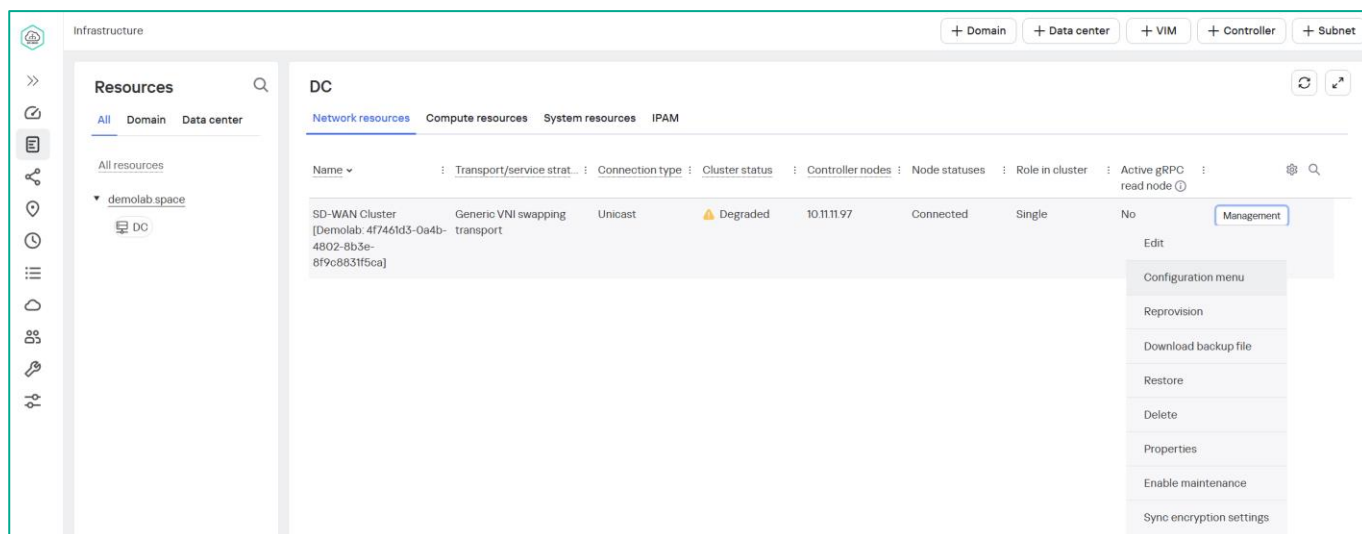
Need update 0

DPID	Model	Switch type	SW version	Name	Role	Status	State	Connection
<input type="checkbox"/> 8000005056AA9EA5	vKESR-M2	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vGW-11	Gateway	Registered	Enabled	Connected

The vGW-11 configuration is complete.

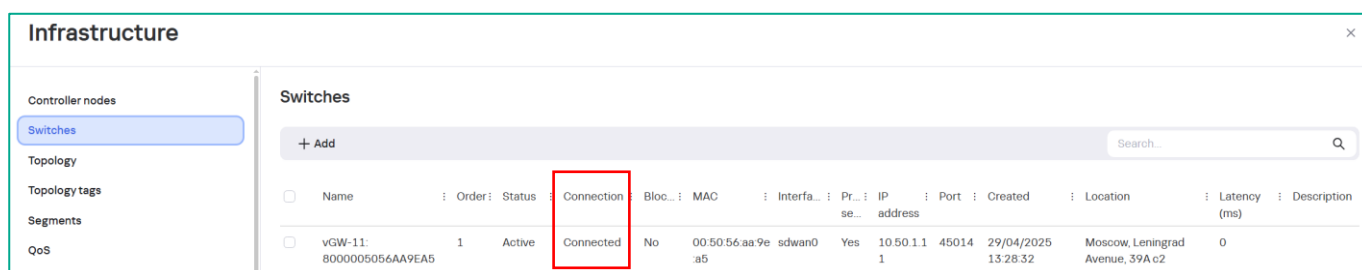
#### 4.9.3. Check the connection of the vGW-11 to the controller.

On the administrator portal go to the **Infrastructure → Domain → DC → Network Resources → SD-WAN cluster → Management → Configuration menu**.



Open **Switches** menu.

Check the vGW-11 gateway connection status. The device status should be **Connected**.

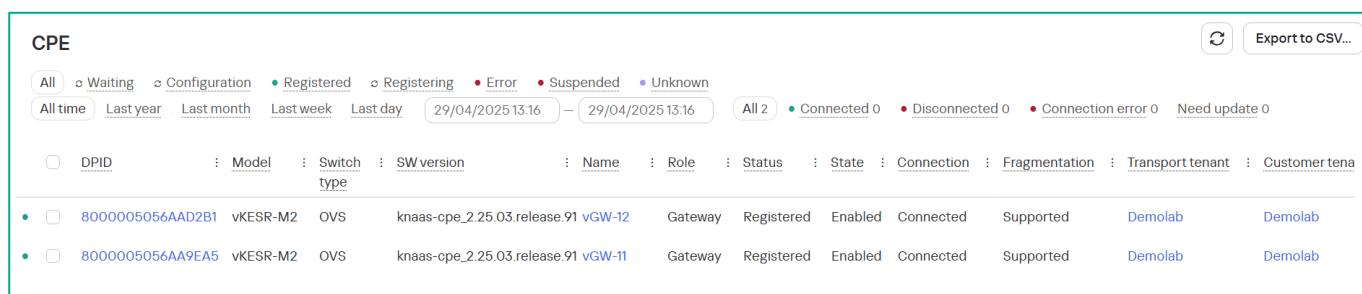


#### 4.9.4. Register vGW-12.

Repeat steps 4.9.1 - 4.9.3 for vGW-12.

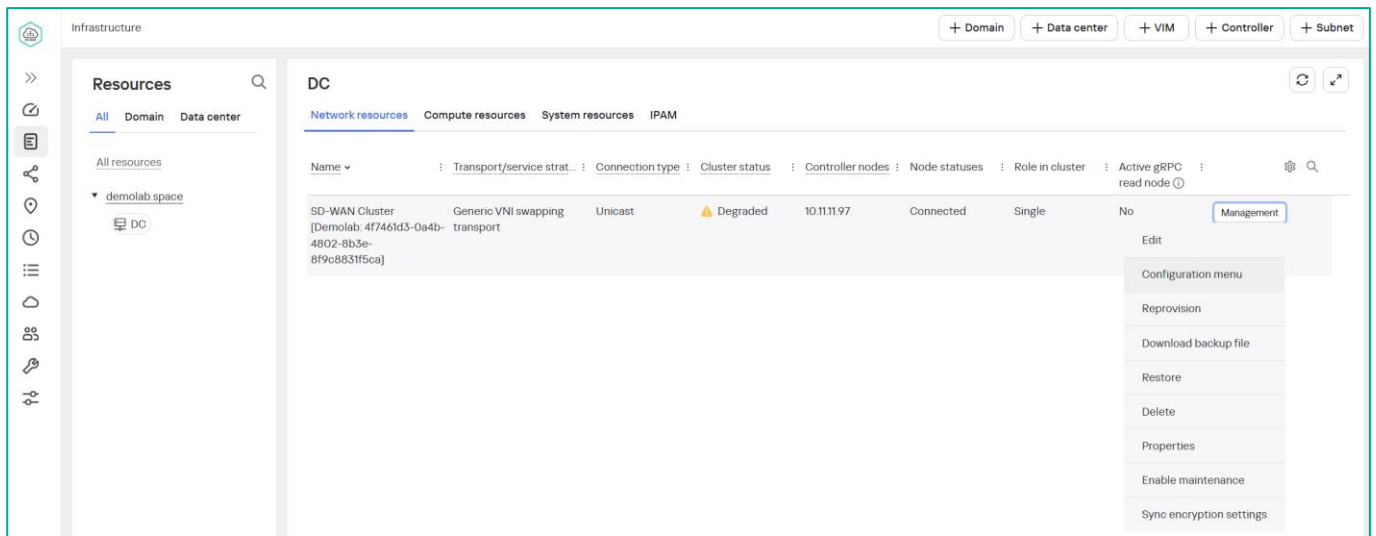
The gateways' connection state should be **Connected** and the status should be **Registered**.

SD-WAN gateways registration is successfully completed.

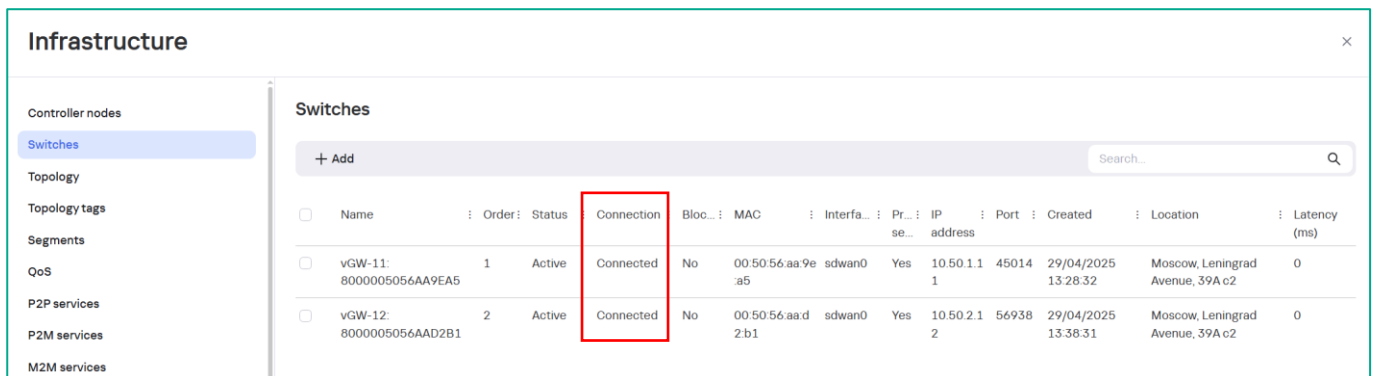


#### 4.9.5. Verify the establishment of GENEVE tunnels between gateways.

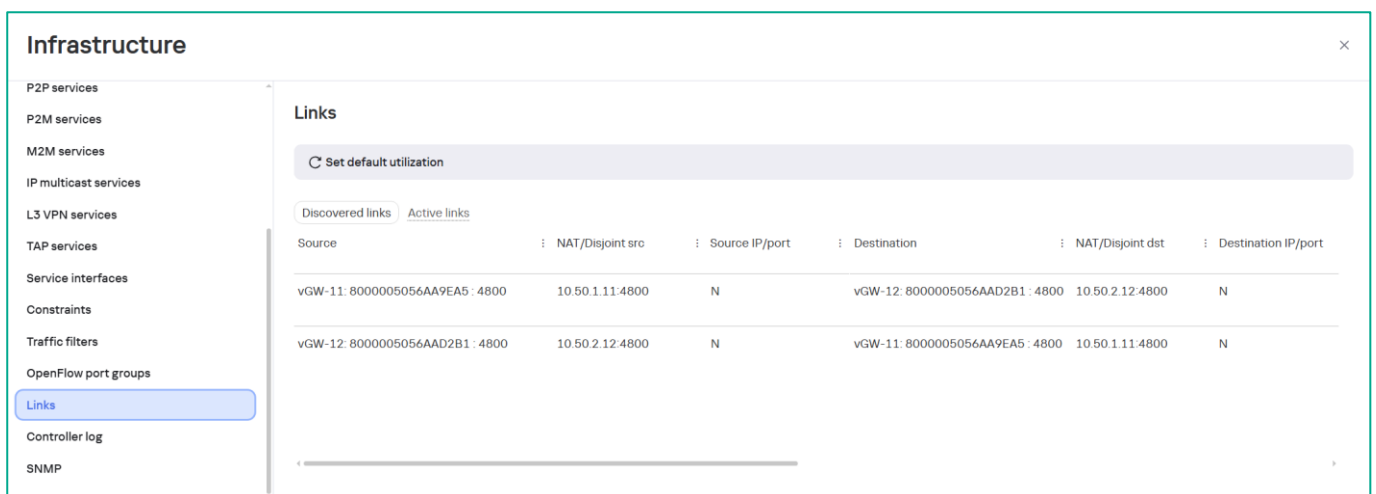
On the administrator portal go to the **Infrastructure → Domain → DC → Network Resources → SD-WAN cluster → Management → Configuration menu**.



Open **Switches** menu and check the OVS (Open vSwitch) status for SD-WAN gateways



Open **Links** menu and check the established GENEVE tunnels between SD-WAN gateways.



#### 4.9.6. Viewing the CPE password for SSH connection.

After the gateways are registered, the orchestrator will change the passwords on the devices. To view the new password, go to the **SD-WAN → CPE**, select the gateway and then click **Show password**. Default user is **root**.

Buttons: Delete, Set location, Disable, **Show password**, Get configuration URL, Update firmware, Unregister, Open SSH console, ⋮

Name: vGW-11

DPID: 8000005056AA9EA5

Location: Бизнес-центр "Олимпия Парк", 39А с2, Leningrad Avenue, Voykovsky District, Moscow,

Transport tenant: Demolab

Customer tenant: Demolab

UNI template: [dropdown]

CPE template: vGW-11

Description: [text area]

NetFlow template: Default NetFlow template

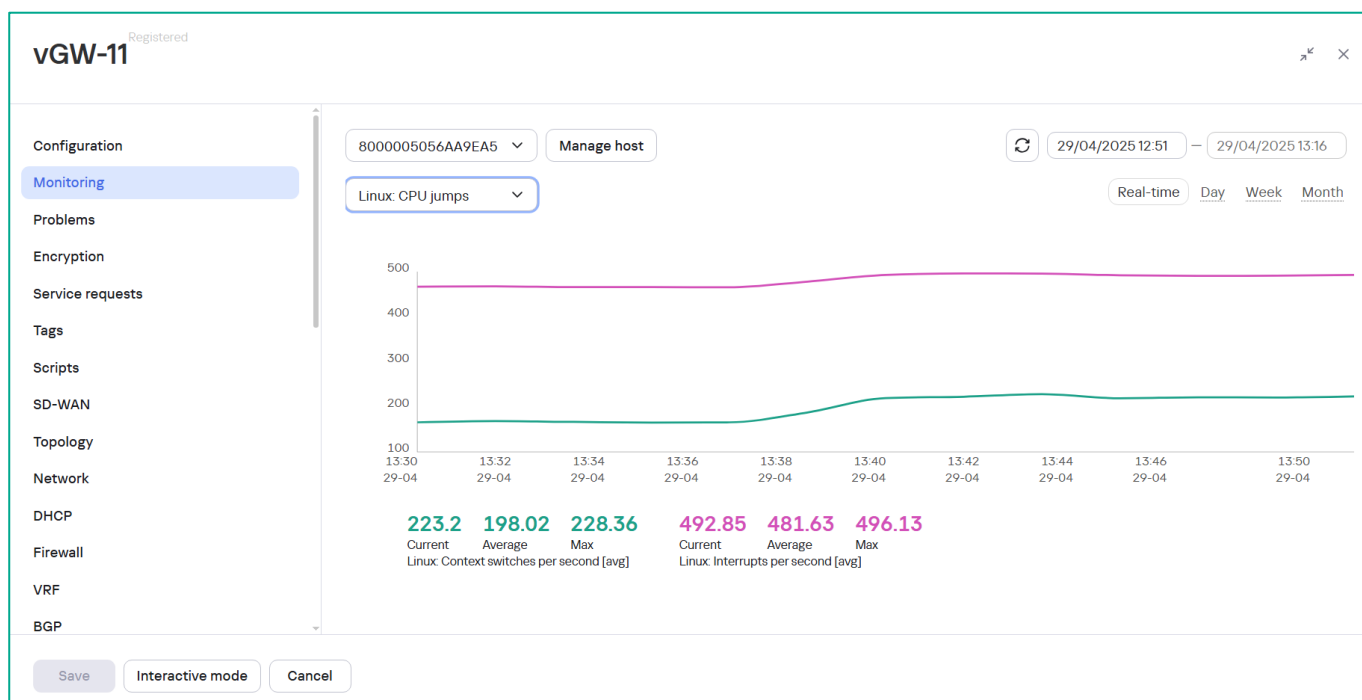
Firewall template: gateway\_firewall\_template (De...

#### 4.9.7. Check the operation of the monitoring system.

Go to the **SD-WAN → CPE**, select SD-WAN gateway and switch to the **Monitoring** tab.

Device statistics will be displayed.

CPE monitoring data is displayed, it may take a while for the interfaces to be detected by the monitoring system and accumulate data for display.

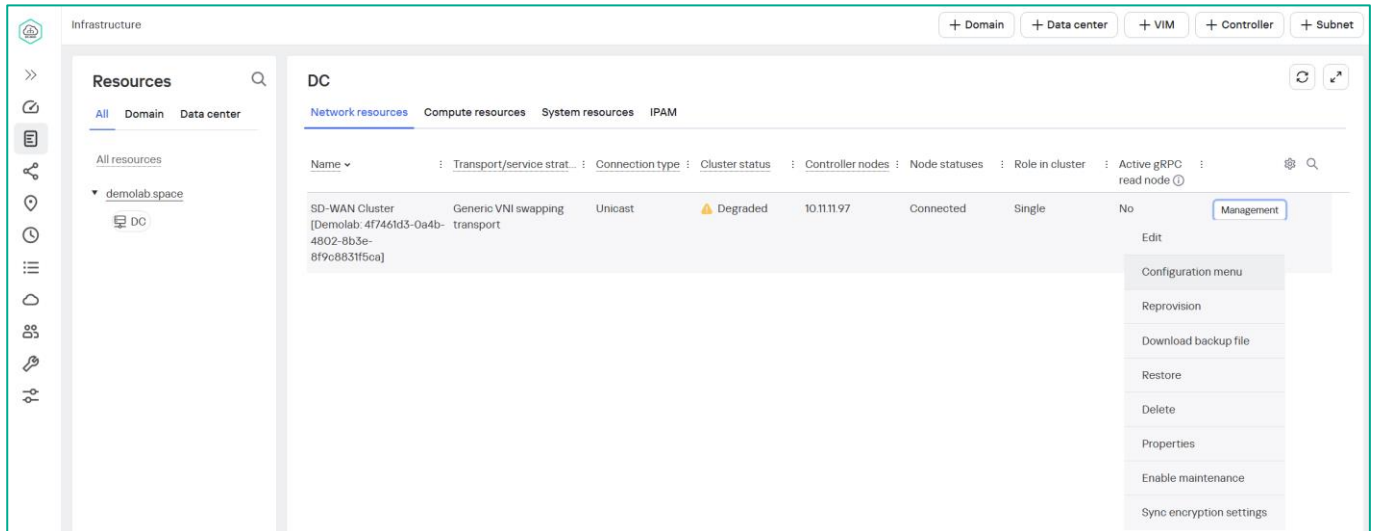


## 4.10. Configuring the Management P2M Service

CPE devices are automatically added to the system's P2M transport service, which is used for device management, specifically the SSH web console.

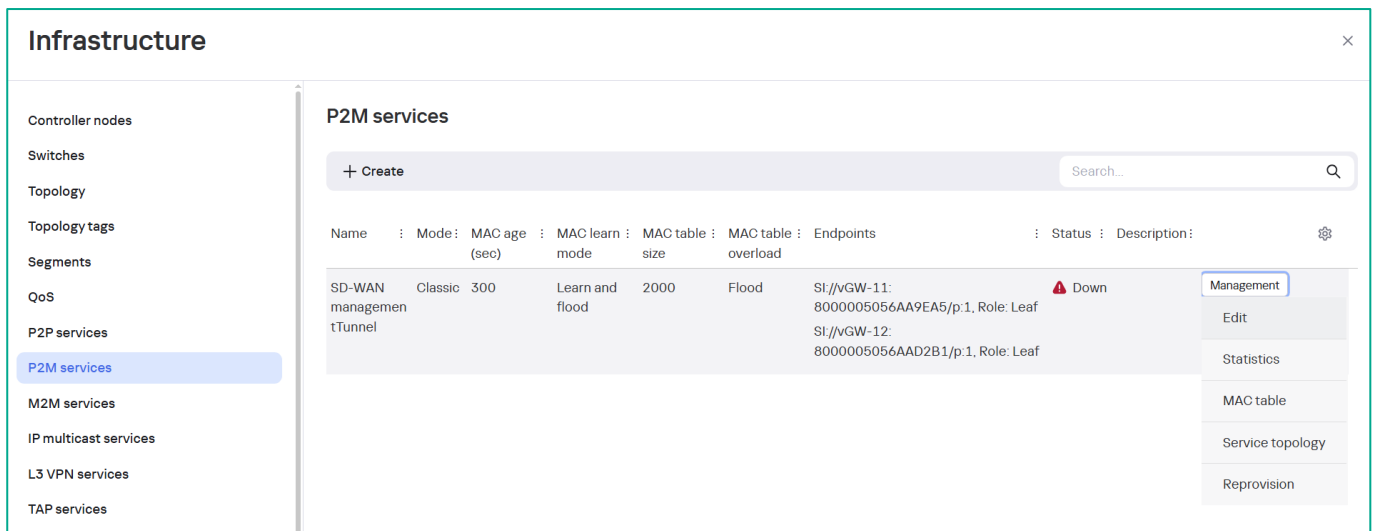
4.10.1. Open P2M service for editing.

Go to **Infrastructure** → **Domain** → **DC** → **Network Resources** → **SD-WAN cluster** → **Management** → **Configuration** menu.

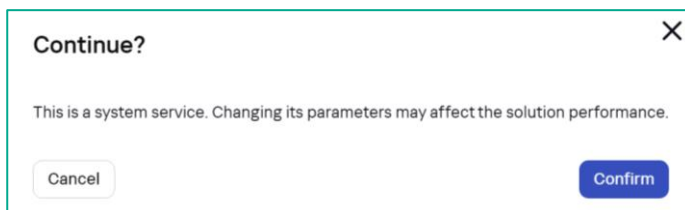


Open **P2M Services** menu.

Select **SD-WAN managementTunnel**, then click **Management** → **Edit**.



**Confirm** opening of the system service for editing.



#### 4.10.2. Set parameters of management P2M transport service.

Click **Next**.

**P2M service** ×

Name: SD-WAN managementTunnel

Constraint: Threshold [ ] ×

Balancing mode ⓘ: Per-flow

Mode: Classic

MAC learn mode: Learn and flood

MAC age (sec): 300

MAC table overload: Flood

MAC table size: 2000

Description: [ ]

Cancel Back Next

Point-to-Multipoint (E-tree in the MEF classification, hereinafter also referred to as P2M service) is a transport service in which traffic is transmitted between multiple service interfaces in accordance with a tree topology. To each service interface added to the P2M service, you must assign one of the following roles:

- Root means the service interface can send traffic to service interfaces with any role.
- Leaf means the service interface can send traffic only to service interfaces with the Root role.

In PoC, the connectivity between the CPE management network and the orchestrator is provided by gateways, so the gateway management service interfaces must be assigned with the Root role. Also, without interfaces with the Root role, the P2M transport service will be in the Down state.

Change gateways service interface's role to **Root**.

Click **Next**.

P2M service

Service endpoints

+ Add

Switch	Service interface	QoS rule	Inbound filter	Role	Backup switch	Backup service interface
vGW-11: 8000005056AA9E...	Port 1, Access	Unlimited-QoS	—	Root	—	—
vGW-12: 8000005056AAD...	Port 1, Access	Unlimited-QoS	—	Root	—	—

Cancel

Back

Next

Click **Save**.

P2M service

Port groups

+ Add

Cancel

Back

Save

**SD-WAN managementTunnel** service configuration is completed.

The service status will change to **UP**.

Infrastructure

Controller nodes

Switches

Topology

Topology tags

Segments

QoS

P2P services

P2M services

M2M services

IP multicast services

P2M services

+ Create

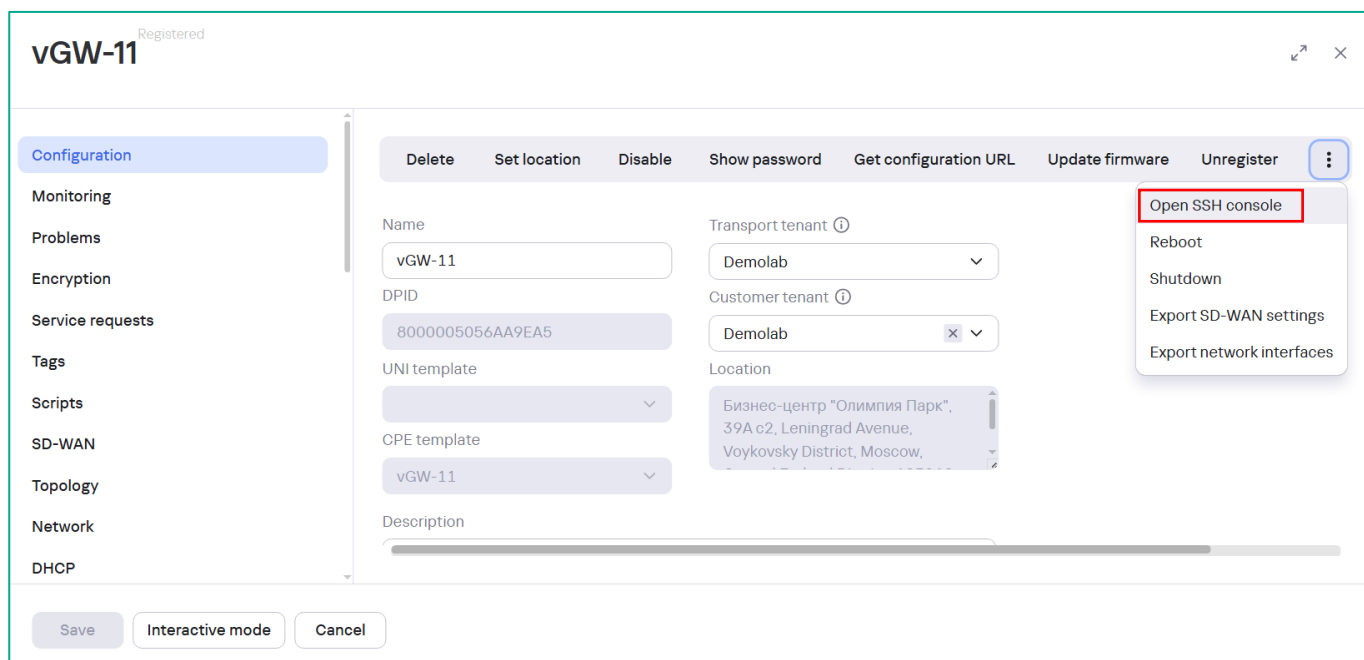
Search...

Name	Mode	MAC age (sec)	MAC learn mode	MAC table size	MAC table overload	Endpoints	Status	Description
SD-WAN management Tunnel	Classic	300	Learn and flood	2000	Flood	SI://vGW-11: 8000005056AA9EA5/p:1, Role: Root SI://vGW-12: 8000005056AAD2B1/p:1, Role: Root	Up	Management

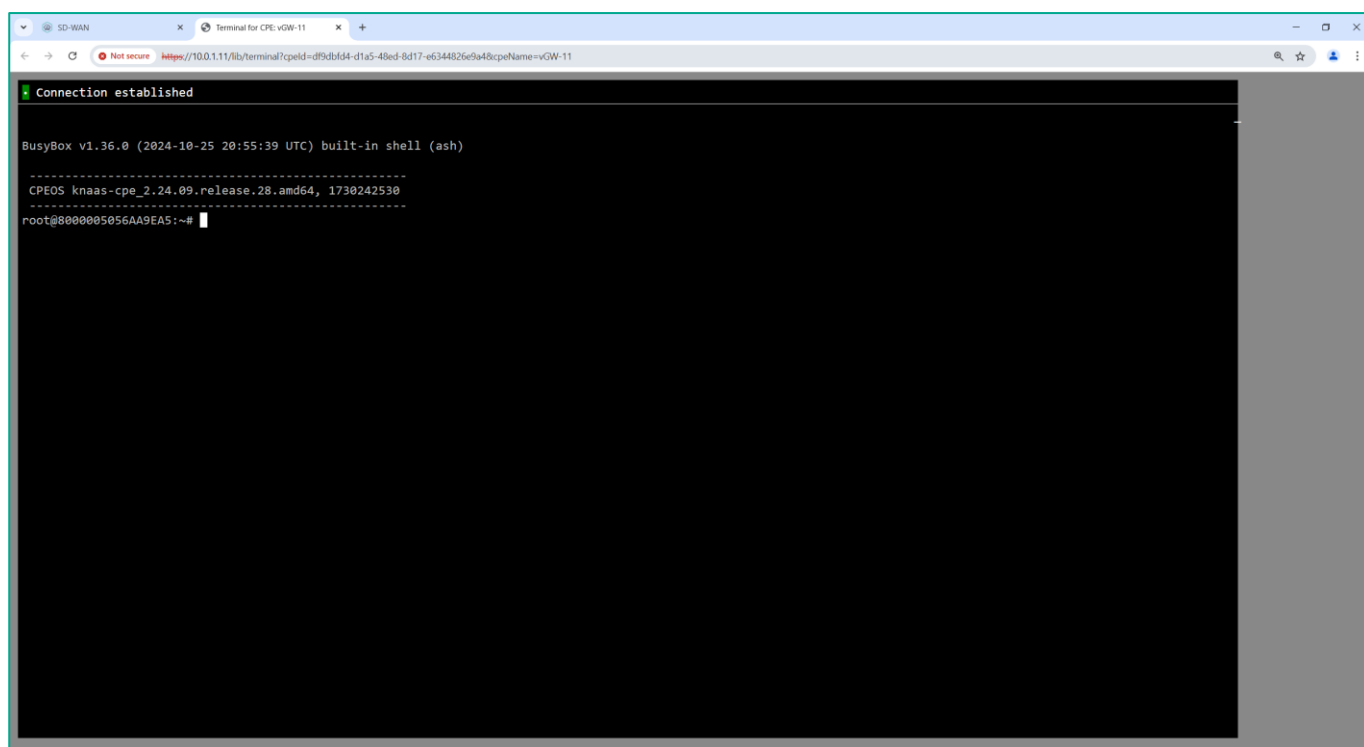


4.10.3. Verify SSH console access to the SD-WAN gateway from the orchestrator web interface.

Go to the **CPE** and select SD-WAN gateway, then click **Open SSH Console**



The SD-WAN gateway console opens in the browser tab.



## 4.11. CPEs initial configuration

4.11.1. Deploy VM **VCPE-3** from CPE vKESR-M1 image (knaas-cpe.<release\_name>.combined.amd64-vkesr-m1.vKESR-M1-esxi.tar.gz).

### Deploy OVF Template

- Select an OVF template**
- Select a name and folder
- Select a compute resource
- Review details
- Select storage
- Ready to complete

### Select an OVF template

Select an OVF template from remote URL or local file system

Enter a URL to download and install the OVF package from the Internet, or browse to a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

☐ URL

☒ Local file

UPLOAD FILES 4 files

CANCEL NEXT

Virtual machine resources for the vKESR-M1:

- 2 x CPU.
- 512 Mb RAM.

### Deploy OVF Template

- Select an OVF template
- Select a name and folder
- Select a compute resource
- Review details**
- Select storage
- Select networks
- Customize template
- Ready to complete

### Review details

Verify the template details.

The OVF package contains advanced configuration options, which might pose a security risk. Review the advanced configuration options below. Click next to accept the advanced configuration options.

Publisher	No certificate present
Product	vCPE
Vendor	Kaspersky Lab
Description	vKESR-M1 vCPU - 2 vRAM - 512M knaas-cpe_2.24.09.release.23.combined.amd64-vkesr-m1
Download size	Unknown
Size on disk	Unknown (thin provisioned) 1.0 GB (thick provisioned)
Extra configuration	guestinfo.urlactivated = false nvram = ovf:/file/file2

CANCEL BACK NEXT

Assign networks according to the PoC topology in Figure 2.

The screenshot shows an example for vCPE-3.

- **WAN1: ISP5.**
- **WAN2: ISP6.**
- **LAN1: cpe3-lan.**

Deploy OVF Template

1 Select an OVF template

2 Select a name and folder

3 Select a compute resource

4 Review details

5 Select storage

**6 Select networks**

7 Customize template

8 Ready to complete

Select networks

Select a destination network for each source network.

Source Network	Destination Network
LAN1	SD-WAN Network
WAN1	ISP5
WAN2	ISP6

3 items

IP Allocation Settings

IP allocation: Static - Manual

IP protocol: IPv4

CANCEL BACK NEXT

Repeat the previous steps and deploy the vCPE-4,vCPE-51,vCPE-52 virtual machine from the vKESR-M1 image.

#### 4.11.2. Configure lan network interface

Open the console to the CPE virtual machines.

The vi text editor is installed on the SD-WAN gateway:

- Press the i key to enter edit mode.
- Press the Esc key to return to command mode.
- :wq - to save the changes and exit.
- :q! - to exit without changes.

Edit the network configuration file:

```
vi /etc/config/network
```

It is necessary to configure the lan network interface to apply the configuration URL from the mgmt workstation.

Change the IP address of vCPE's **lan** interface according to Table 1 in section 2.2.

The screenshot shows an example for vCPE-3.

```
config interface 'ovs_lan'
    option device 'ovs-lan'
    option proto 'none'

config interface 'lan'
    option type 'bridge'
    option proto 'static'
    option ipaddr '10.20.3.1'
    option netmask '255.255.255.0'
    option ifname 'eth2'
    option auto '1'
    option force_link '1'
```

Restart network service:

```
/etc/init.d/network restart
```

Check the applied settings:

```
ip -br a
```

```
root@8000005056AAC4FD:~# ip -br a
lo                UNKNOWN          127.0.0.1/8 ::1/128
eth0              UP                10.50.5.9/24 fe80::250:56ff:feaa:c4fd/64
eth1              UP                10.50.6.15/24 fe80::250:56ff:feaa:abf2/64
eth2              UP
ip6tnl0@NONE      DOWN
gre0@NONE         DOWN
gretap0@NONE      DOWN
erspan0@NONE      DOWN
ip6gre0@NONE      DOWN
bond0             DOWN
br-lan            UP                10.20.3.1/24
overlay@ovs-lan   UP
ovs-lan@overlay   UP
mgmt@ovs-mgmt     UP
ovs-mgmt@mgmt     UP
```

After registration (applying the configuration URL in step 4.14.2) CPE will receive and replace the network settings according to the settings in 4.13.5.

**Note:** You can also apply the configuration URL when deploying CPEs from OVF. To obtain URL, you must first create a CPE using previously created gateway templates with a random DPID and copy the configuration URL (described in step 4.14.2). When configuring the CPE, you must use default IP address for the Configuration URL - 192.168.7.1. After booting, the CPE will appear in the orchestrator with the status Unknown. Next, open the CPE, click Register in the Configuration menu, and then configure the CPE settings as described in step 4.14.

## 4.12. Creating firewall template for CPE devices

Workstations connected to the CPE must be able to access public networks. This requires creating an additional WAN zone with masquerading enabled, and creating a template with configured forwarding between the lan zone and the wan zone.

### 4.12.1. Create an additional firewall zone for CPE devices.

Connect to the tenant self-service portal (PoC uses the tenant **Demolab**), to do this, click **Connect as Tenant** from the **Tenants** menu or connect to the SD-WAN orchestrator by the administrator of the created tenant.

**Note:** When zones and firewall templates are created by an administrator from the administrator portal, they will not be available to users with tenant roles.

Go to **SD-WAN > Firewall zones**.

Name	Usage	Author	Created
lan	Yes	admin (Demolab)	23/04/2025 15:57:47
wan	Yes	admin (Demolab)	23/04/2025 15:57:47
mgmt	No	admin (Demolab)	23/04/2025 15:57:47
mgmt_gw	Yes	admin (Demolab)	29/04/2025 11:17:34

Click **+ Firewall Zone**.

Enter the new firewall zone **Name: wan\_cpe**.

Check **Masquerading**.

Click **Create**.

**New firewall zone**

Name: wan\_cpe

Input: ACCEPT Output: ACCEPT Forwarding: REJECT

☒ Masquerading ☒ MSS clamp to PMTU ☐ Drop logging

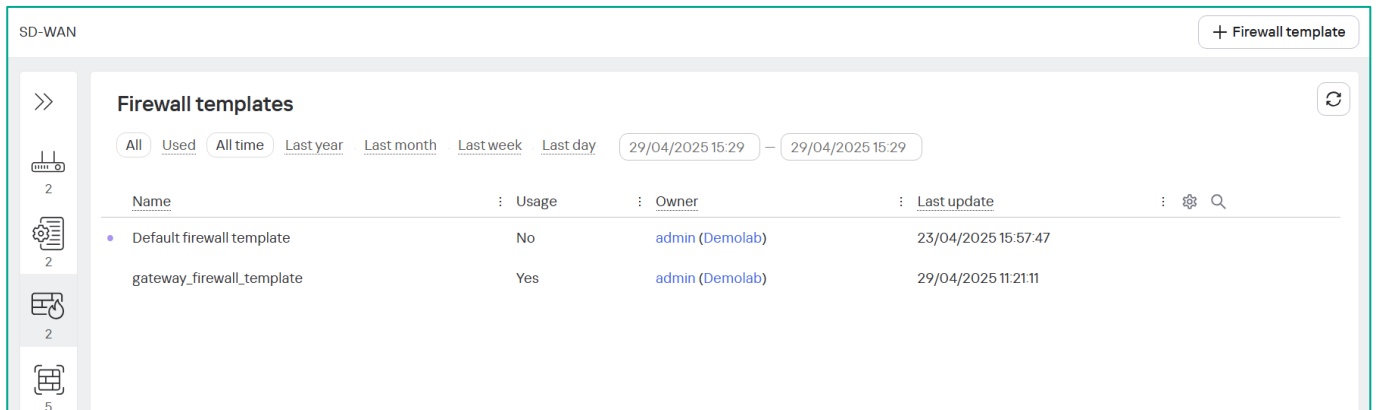
Masquerading source subnets: + Add Masquerading destination subnets: + Add

Networks: + Add

Create Cancel

## 4.12.2. Create the CPE firewall template.

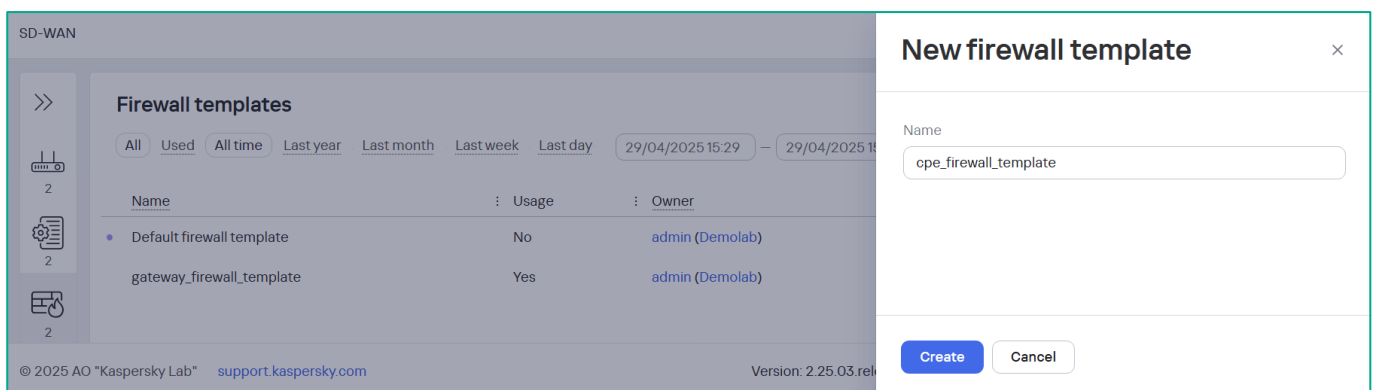
Go to **SD-WAN > Firewall templates**.



Click **+ Firewall Template**.

Click the new firewall template **Name: cpe\_firewall\_template**.

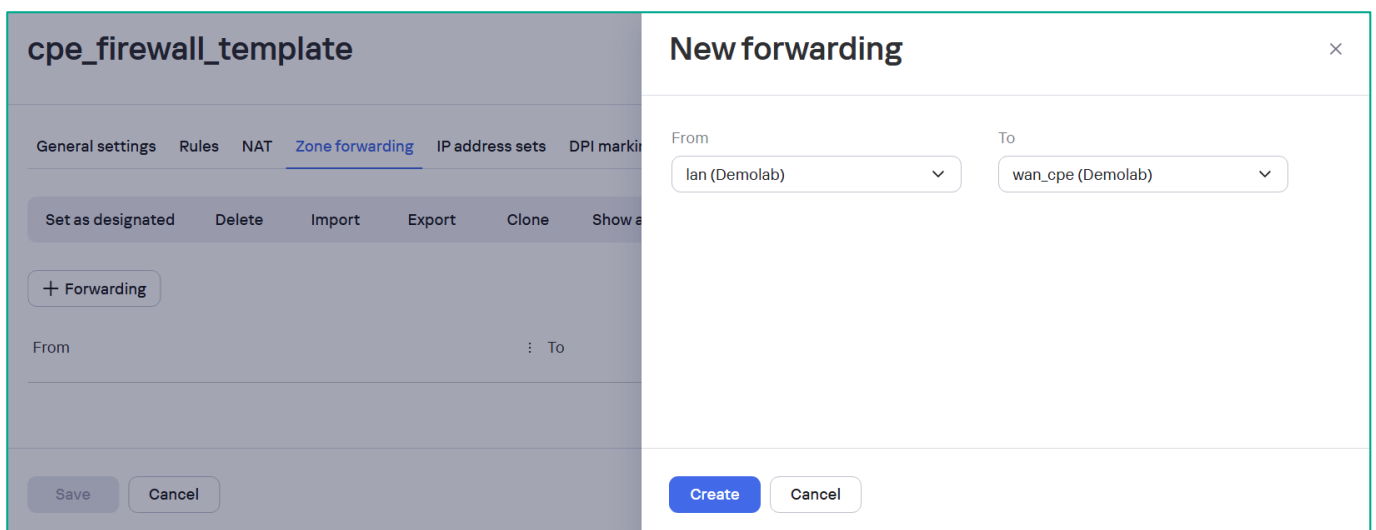
Click **Create**.



Switch to the **Zones forwarding** tab.

Create a new forwarding rule from zone **lan** to **wan\_cpe**.

Click **+ Forwarding** and select the required zones.



Click **Save**.

cpe\_firewall\_template

General settingsRulesNATZone forwardingIP address setsDPI marking

Set as designatedDeleteImportExportCloneShow associated CPEs

+ Forwarding

From	To	Actions
lan (Demolab)	wan_cpe (Demolab)	Delete

Save

Cancel

## 4.13. Creating CPE devices templates

### 4.13.1. Create the vCPE-3 device template.

Connect to the tenant self-service portal (PoC uses the tenant **Demolab**), to do this, click **Connect as Tenant** from the **Tenants** menu or connect to the SD-WAN orchestrator by the administrator of the created tenant.

**Note:** When CPE templates are created by an administrator from the administrator portal, they will not be available to users with tenant roles.

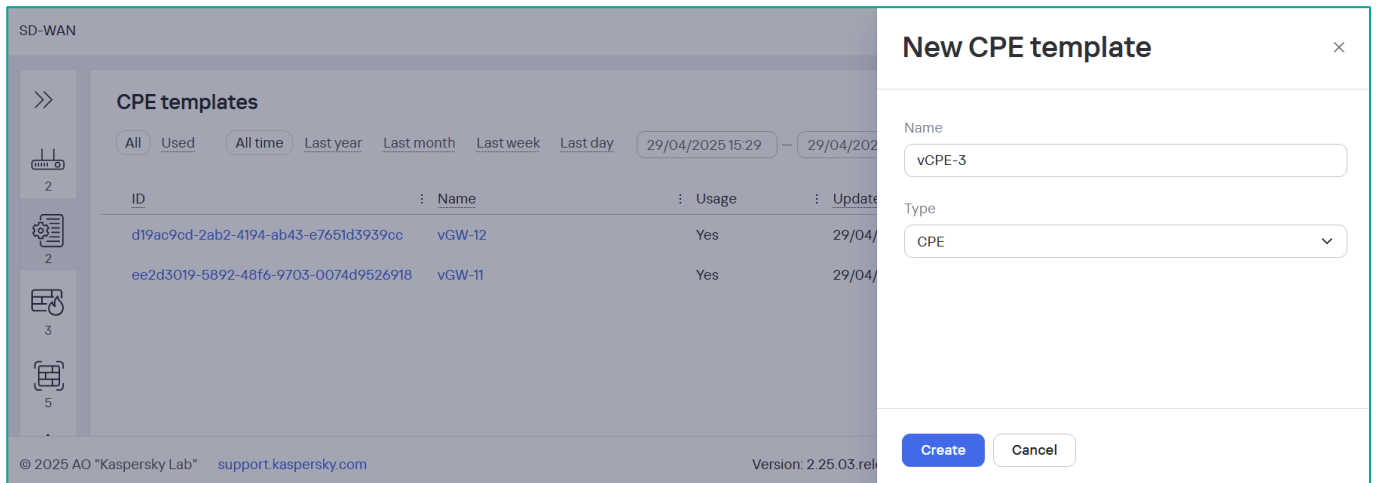
Go to **SD-WAN > CPE Templates**.

Click **+ CPE Template**.

Specify vCPE-3 template parameters:

- **Name:** vCPE-3
- **Type:** CPE

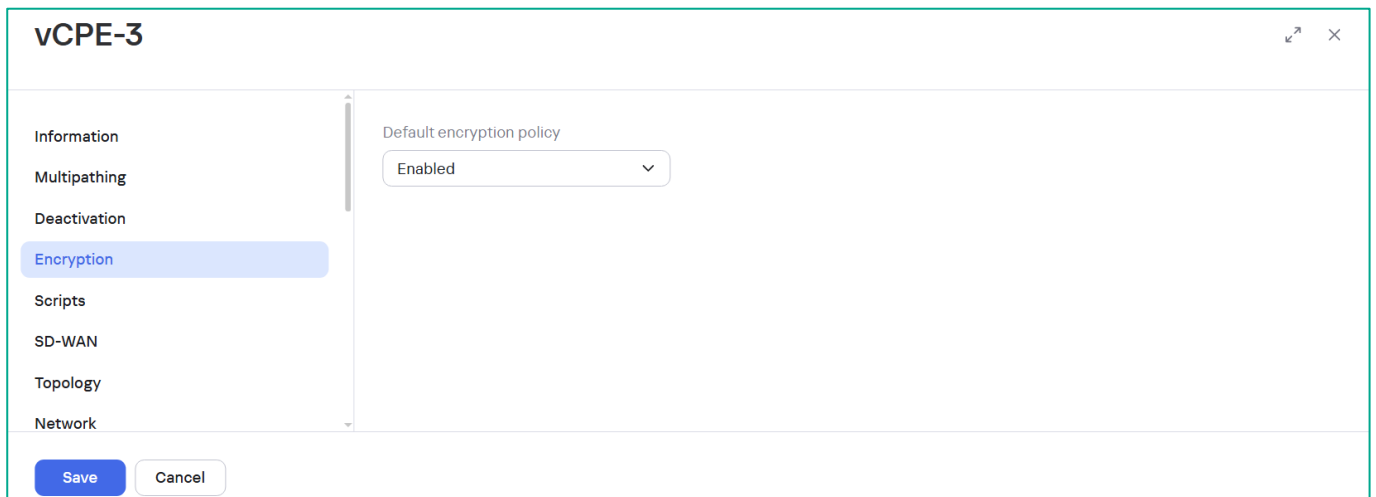
Click **Create**.



### 4.13.2. Set encryption policy in the vCPE-3 template.

Go to the **Encryption** menu.

Set the encryption policy to **Enabled**.





#### 4.13.3. Set SD-WAN parameters in the vCPE-3 template.

Go to **SD-WAN → General settings** menu.

Set the IP address to connect to the orchestrator (in the PoC it is orc1 public address, it is also possible to use a domain name to connect).

Set the following parameters:

- **Orchestrator IP/FQDN: 10.50.1.14.**
- **Orchestrator Port: 443.**
- **Openflow Transport: ssl.**
- **Control SD-WAN interface: sdwan0.**
- Change IP-address **192.168.7.1** in **Configuration URL** to **10.20.3.1** (lan interface IP address).

The screenshot displays the vCPE-3 configuration window with the 'SD-WAN' tab selected in the left-hand menu. The main configuration area is divided into two sections: 'Connection to orchestrator' and 'Connection to controller'.

**Connection to orchestrator:**

- Orchestrator IP address/FQDN: 10.50.1.14
- Orchestrator port: 443
- ☐ Backup orchestrator IP address and port
- Orchestrator protocol: https
- Update interval (sec): 30
- Interactive update interval (sec): 3
- Interactive mode timeout (sec): 180

**Connection to controller:**

- OpenFlow transport: SSL
- Control SD-WAN interface: sdwan0
- ☐ Preemption
- Auto-reboot: No
- Reboot timeout (sec): 86400
- Configuration URL: http://10.20.3.1/cgi-bin/config?payload={config}

At the bottom of the window, there are 'Save' and 'Cancel' buttons.

4.13.4. Set CPE topology role in the vCPE-3 template.

Go to the **Topology** menu.

Set **Role** to **CPE**.

The screenshot shows the 'vCPE-3' configuration window. On the left is a sidebar menu with the following items: Information, Multipathing, Deactivation, Encryption, Scripts, SD-WAN, Topology (highlighted in blue), Network, DHCP, and BGP. The main area on the right contains the following settings:

- Role** (with an information icon): A dropdown menu currently showing 'CPE'.
- Transit CPE** (with an information icon): An unchecked checkbox.
- Topology tags** (with an information icon): A large empty text input field.

At the bottom of the window are two buttons: 'Save' (in blue) and 'Cancel' (in white).

## 4.13.5. Configure network interfaces in the vCPE-3 template.

Go to the **Network** menu.

Add the following network interfaces (click **+ Network interface**):

- **sdwan0: eth0.**
- **sdwan1: eth1.**
- **lan: eth1.**
- **overlay: overlay.**

Below is a description of the interface parameters.

vcPE-3

Information

Multipathing

Deactivation

Encryption

Scripts

SD-WAN

Topology

Network

DHCP

BGP

VRF

OSPF

Routing filters

PBR

BFD

+ Network interface

Alias	Zone	Interface name	Protocol	IP address/mask	MTU	Enable automatically	Actions
lan	lan	eth2	Static IPv4 address	IP address: 10.20.3.1 Mask: 255.255.255.0		Yes	<a>Edit</a> <a>Delete</a> <a>Disable</a>
overlay	lan	overlay	Static IPv4 address	IP address: 172.16.1.3 Mask: 255.255.255.0		Yes	<a>Edit</a> <a>Delete</a> <a>Disable</a>
sdwan0	wan_cpe	eth0	DHCP client			Yes	<a>Edit</a> <a>Delete</a> <a>Disable</a>
sdwan1	wan_cpe	eth1	DHCP client			Yes	<a>Edit</a> <a>Delete</a> <a>Disable</a>

Save

Cancel

Add the lan network interface with following parameters:

- **Alias:** lan.
- **Zone:** lan.
- **Interface name:** eth1.
- **Protocol:** Static IPv4 address.
- **IPv4 address:** 10.20.3.1/24.

Click **Create**.

The screenshot shows the 'New network interface' configuration window. The 'Alias' field is set to 'lan', the 'Zone' is 'lan (Demolab)', and the 'Interface name' is 'lan'. The 'Bridge' and 'NetFlow' checkboxes are unchecked. The 'Protocol' is set to 'Static IPv4 address'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'IPv4 address and subnet mask input type' is set to 'Manually'. The 'IPv4 address' field contains '10.20.3.1' and the 'IPv4 netmask' field contains '255.255.255.0'. At the bottom, there are 'Create' and 'Cancel' buttons.

Add the overlay network interface with following parameters:

- **Alias:** overlay.
- **Zone:** lan.
- **Interface name:** overlay.
- **Protocol:** Static IPv4 address
- **IPv4 address:** 172.16.1.3/24.
- Check **Generate MAC address automatically**. With this setting, the interface's MAC address is automatically generated from the pool and saved after the device reboots, which prevents neighboring devices from learning the MAC addresses of overlay interfaces and speeds up the convergence time of routing protocols.

Click **Create**.

The screenshot shows the 'New network interface' configuration window for an overlay interface. The 'Alias' field is set to 'overlay', the 'Zone' is 'lan (Demolab)', and the 'Interface name' is 'overlay'. The 'Bridge' and 'NetFlow' checkboxes are unchecked. The 'Protocol' is set to 'Static IPv4 address'. Under 'Settings', 'Enable automatically' is checked, and 'Force IP address, route, and gateway' is unchecked. The 'IPv4 address and subnet mask input type' is set to 'Manually'. The 'IPv4 address' field contains '172.16.1.3' and the 'IPv4 netmask' field contains '255.255.255.0'. There are also fields for 'IPv4 gateway' and 'IPv4 broadcast', both of which are empty. Below these, there is a 'DNS servers' section with an '+ Add' button. At the bottom, there are fields for 'Override MAC', 'Override MTU', and 'Route metric', all of which are empty. The 'Generate MAC address automatically' checkbox is checked. At the bottom, there are 'Create' and 'Cancel' buttons.

Add the **sdwan0** and **sdwan1** network interfaces:

- **Alias:** sdwan0 / sdwan1.
- **Zone:** wan\_cpe.
- **Interface name:** eth0.
- **Protocol:** DHCP client.

Click **Create**.

#### 4.13.6. Enable DHCP server on the lan interfaces.

You must add a DHCP Server on the interface so that workstations can obtain IP addresses via DHCP.

Go to the **DHCP** menu.

Click **+ DHCP Server**.

Set the parameters of the DHCP server:

- **Network interface alias:** lan.
- **Type:** Server.
- **Start IP** (first IP of the DHCP range): **51**.
- **Limit** (DHCP range size): **50**.
- **Lease time:** **12 hours**.
- **DNS servers:** **8.8.8.8**.

Click **Create**.

The screenshot shows the 'vcPE-3' configuration window with the 'Network' section selected. The 'DHCP' sub-section is highlighted in the left sidebar. A '+ DHCP server' button is visible. The 'New DHCP server' dialog box is open, showing the following configuration:

Field	Value
Network interface alias	lan
Network interface IP/mask	10.20.3.0/24
Type	<input type="radio"/> Disabled <input type="radio"/> Relay <input checked="" type="radio"/> Server
Start IP ①	51
Limit ①	50
DHCP range	10.20.3.51 - 10.20.3.100
Lease time ①	12 hours
<b>DHCP options</b>	
IPv4 gateway (3) ①	
DNS servers (6) ①	8.8.8.8
NTP servers (42) ①	
Custom option ①	+ Option

At the bottom of the dialog box are 'Create' and 'Cancel' buttons. The background window has 'Save' and 'Cancel' buttons at the bottom left.

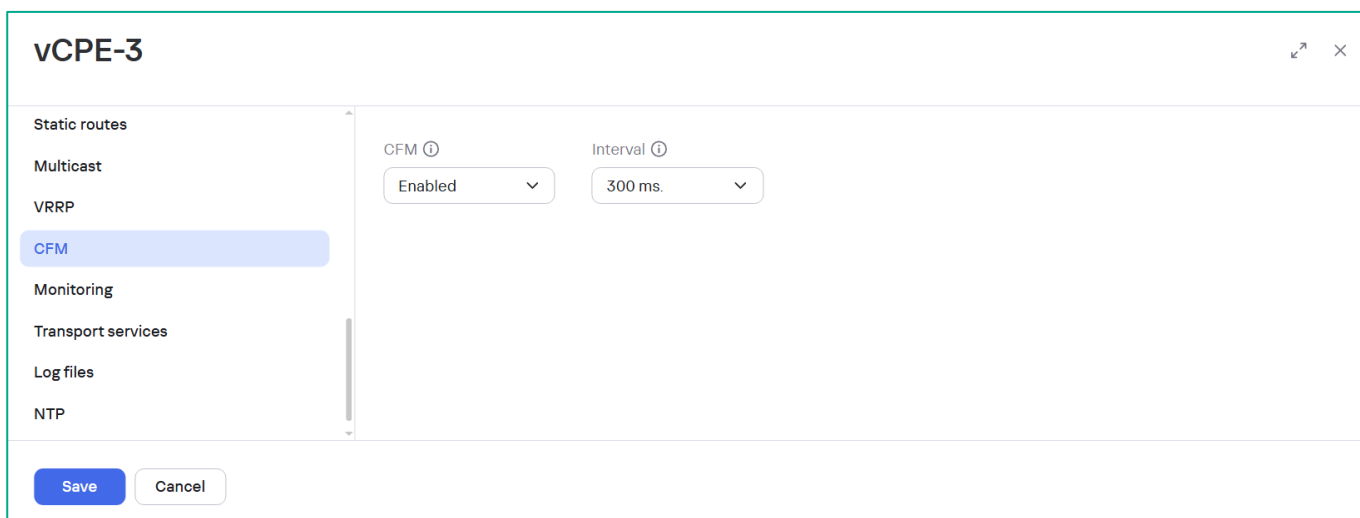
#### 4.13.7. Configure CFM in the vCPE-3 template.

The Connectivity Fault Management functionality allows detecting unavailable links between CPE devices. When CFM is enabled, the CPE device sends Continuity Check Message control packets over its links at the specified intervals, and listens for response control packets on opposite-direction links. If response control packets do not arrive, the CPE device considers the link unavailable and starts transmitting traffic over a randomly selected available link.

Go to the **CFM** menu.

Set the following parameters:

- **CFM: Enabled** (enable CFM for all links).
- **Interval: 300 ms** (interval for sending control packets).



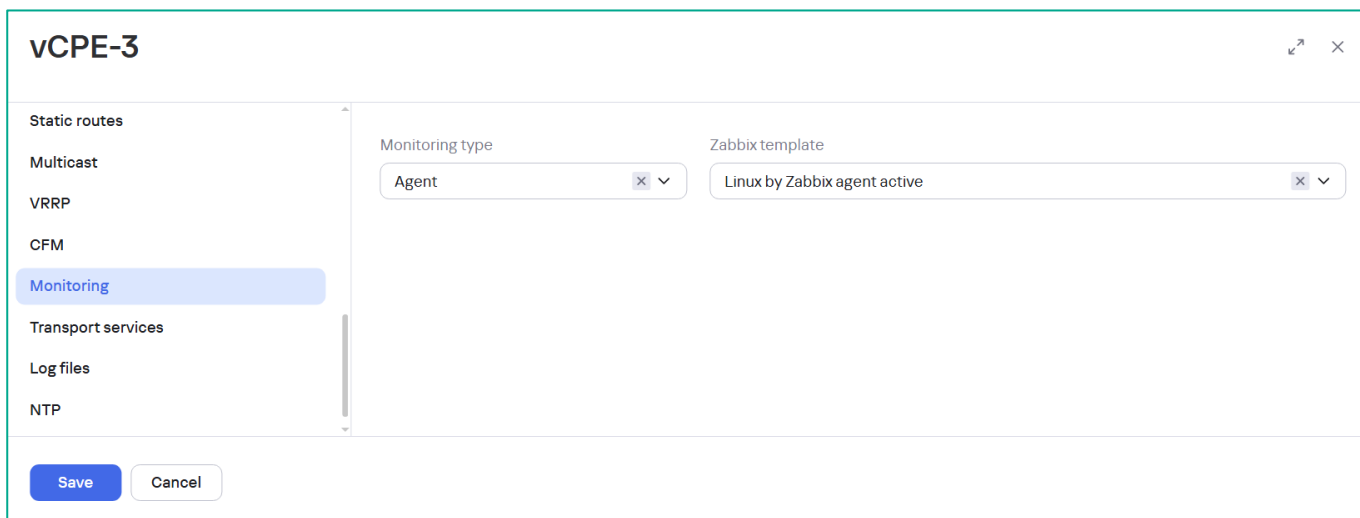
The screenshot shows the 'vCPE-3' configuration window. On the left, a sidebar lists various configuration categories: Static routes, Multicast, VRRP, CFM (highlighted in blue), Monitoring, Transport services, Log files, and NTP. The main area displays the 'CFM' configuration. It features two dropdown menus: 'CFM' set to 'Enabled' and 'Interval' set to '300 ms'. At the bottom, there are 'Save' and 'Cancel' buttons.

#### 4.13.8. Configure monitoring parameters in the vCPE-3 template.

Go to the **Monitoring** menu.

Set the following parameters:

- **Monitoring type: Agent.**
- **Zabbix template: Linux by Zabbix agent active.**



The screenshot shows the 'vCPE-3' configuration window with the 'Monitoring' menu selected in the sidebar. The main area displays the 'Monitoring' configuration. It features two dropdown menus: 'Monitoring type' set to 'Agent' and 'Zabbix template' set to 'Linux by Zabbix agent active'. At the bottom, there are 'Save' and 'Cancel' buttons.

## 4.13.9. Configure NTP in the vCPE-3 template.

Go to the **NTP** menu.

By default, the NTP client is enabled and pool.ntp.org is used.

## 4.13.10. Create Prefix List in the vCPE-3 template.

Switch to the **Routing filters → Prefix lists** tab.

Click **+ Prefix List**.

Set **Name**: **cpe-lan**.

Click **+ Rule**.

Add the following prefix:

- **Seq: 10.**
- **IP/mask: 10.20.0.0/16.**
- **Greater or Equal: 17.**

Click **Create**.



4.13.11. Create route map in the vCPE-3 template.

Switch to the **Routing filters** → **Route maps** tab.

Click **+ Route Map**.

Set **Name**: **cpe-route-map**.

Click **+ Rule** and set rule parameters:

- **Sequence**: **10**.
- **Action**: **Permit**.
- **Match Type**: **Prefix-list**.
- **Prefix list**: **cpe-lan**.

Click **Create**.

**vCPE-3**

SD-WAN  
Topology  
Network  
DHCP  
BGP  
VRF  
OSPF  
Routing filters

Access control  
+ Route map  
Name

**New route map**

Name ①  
cpe-route-map

+ Rule

Sequence	Action	Match type	Value	Change attribute	New value
10	Permit	Prefix-List	66710700-24fb-11	None	

Prefix list  
cpe-lan

Create Cancel

4.13.12. Configure BGP in the vCPE-3 template.

Go to the **BGP** menu.

Specify BGP **Autonomous System**: **65500**.

Click **+ BGP** to add a new BGP instance.

**vCPE-3**

SD-WAN  
Topology  
Network  
DHCP  
BGP  
VRF  
OSPF  
Routing filters  
PBR  
BFD

Autonomous System  
65500

Default BGP Instance with VRF ①

+ BGP

State	VRF	Router ID	BGP neighbor	Peer groups	Route Distinguisher	Export routes	Import routes	A
No data								

Save Cancel

Switch to the **General settings** tab.

Set the BGP parameters:

- **BGP: Enabled.**
- **Router ID: 172.16.1.3** (overlay interface IP address).
- **Maximum Paths: 2.**
- **Graceful Restart.**
- **Default IPv4 Unicast.**
- **BGP Timers.**
  - **Keepalive: 10.**
  - **Holdtime: 30.**

Apply **Route Map: cpe-route-map** for **Connected** routes redistribution.

## New BGP instance

[General settings](#)
[Neighbors](#)
[Peer groups](#)
[Route leaking](#)

BGP

Enabled

VRF

main/254

AS

65500

Router ID

172.16.1.3

☐ Router ID from IP pool

Maximum paths

2

☐ Always compare MED

☒ Graceful restart (helper mode)

☒ Use default IPv4 unicast routes

☒ BGP timers

Keepalive (sec)

10

Holdtime (sec)

30

Route redistribution

☐ Kernel

Route map

Metric

☒ Connected

Route map

cpe-route-map x

Metric

Save

Cancel

4.13.13. Configure BGP neighborships to vGW-11 and vGW-12 from vCPE-3.

Switch to the **Neighbors** tab and click **+ BGP Neighbor**

Create 2 BGP neighbors. Set the BGP neighbor parameters:

- **Name: vGW-11.**
- **Neighbor IP: 172.16.1.11.**
- **Remote AS: 65500.**
- **Name: vGW-12.**
- **Neighbor IP: 172.16.1.12.**
- **Remote AS: 65500.**

New BGP instance

General settings

Neighbors

Peer groups

Route leaking

+ BGP neighbor

Neighbor IP	Name	Description	Remote AS	Shutdown	Weight	Actions
172.16.1.12	vGW-12		65500	No		<a>Edit</a> <a>Delete</a>
172.16.1.11	vGW-11		65500	No		<a>Edit</a> <a>Delete</a>

Save

Cancel

Click **Save**.

Set **Default BGP Instance with VRF: main/254** (default VRF for BGP instances, used for backward compatibility with older versions of CPE devices).

Click **Save** to save CPE template.

vCPE-3

SD-WAN

Topology

Network

DHCP

BGP

VRF

OSPF

Routing filters

PBR

BFD

Static routes

Autonomous System

65500

Default BGP Instance with VRF

main/254

+ BGP

State	VRF	Router ID	BGP neighbor	Peer groups	Route Distinguisher	Export routes	Import routes	Actions
Enabled	main/254	172.16.1.3	2	0	65500/254	Off	Off	<a>Edit</a> <a>Delete</a>

Save

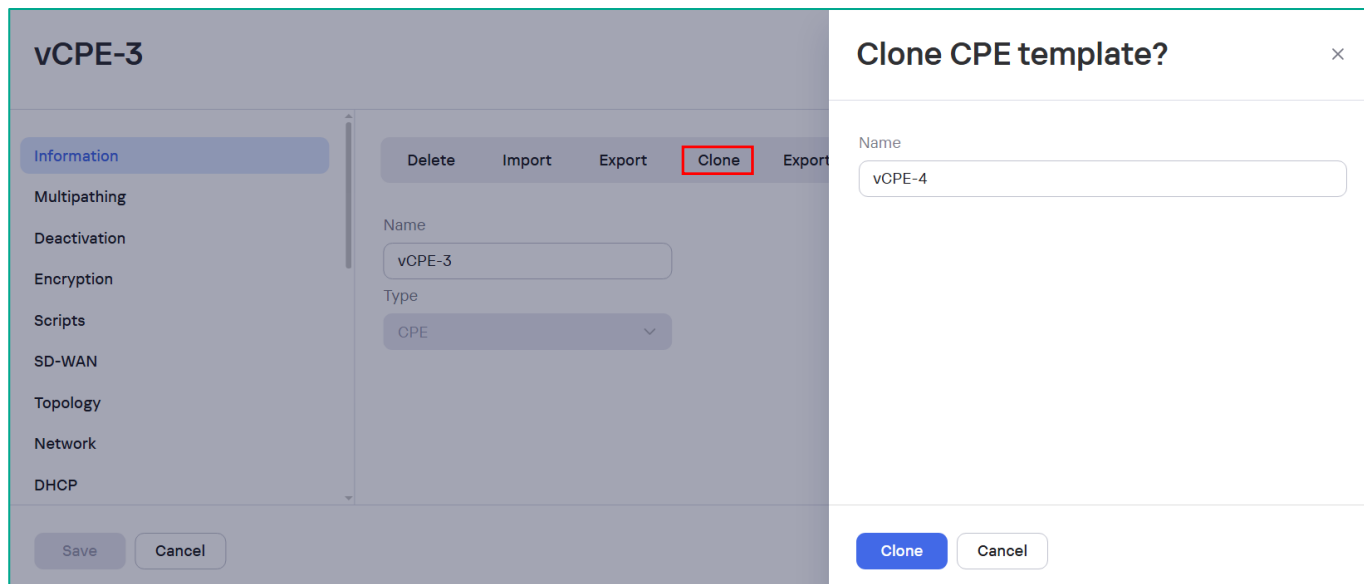
Cancel

4.13.14. Create CPE device templates for vCPE-4, vCPE-51, and vCPE-52 using cloning.

Clone the template for **vCPE-4**, **vCPE-51** and **vCPE-52** from **vCPE-3**.

Open vCPE-3 template, go to the **Information** menu then click **Clone**

Set the new template **Name**.



Repeat steps 4.13.3 - 4.13.13 for the cloned templates, changing the parameter values to match each CPE device according to Table 1.

Change IP addresses for:

- **Configuration URL.**
- **lan** interface.
- **overlay** interface.
- **BGP Router ID.**

It is also required to add a DHCP server similar to section 4.13.6.

## 4.13.15. Configure the VRRP on vCPE-51 and vCPE-52

Kaspersky SD-WAN supports the installation of multiple CPE devices at sites to ensure high availability. One of the options to provide high availability is to use the VRRP (Virtual Router Redundancy Protocol).

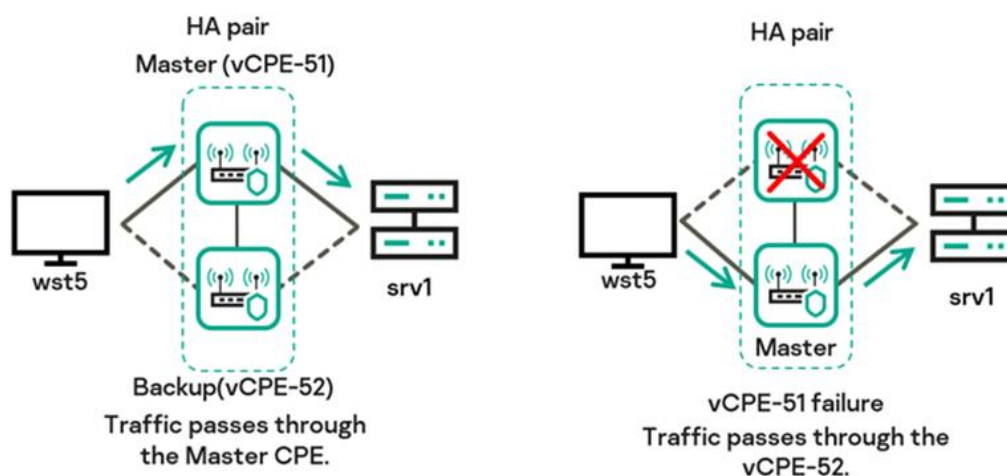


Figure 4 - CPE redundancy with the VRRP protocol

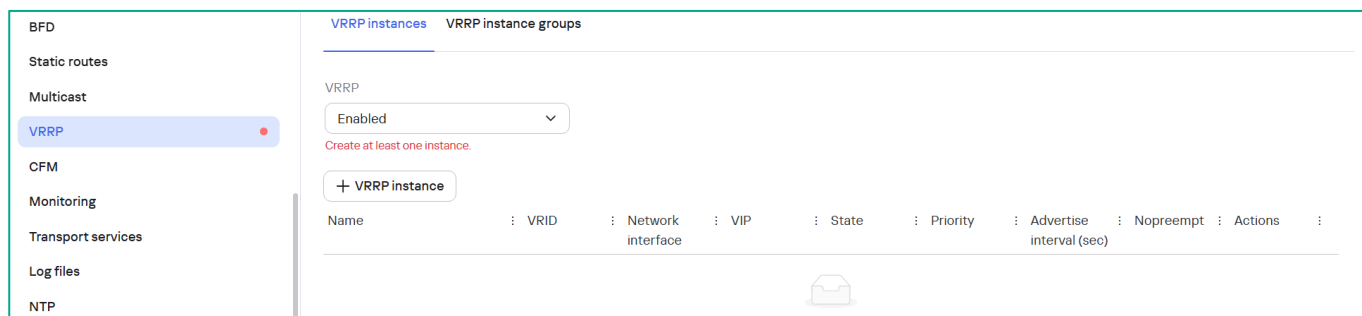
For more information, please refer to Kaspersky SD-WAN Online Help:

<https://support.kaspersky.com/help/SD-WAN/2.4/en-US/246585.htm>

Configure VRRP in the vCPE-51 and vCPE-52 templates.

Open the CPE template and go to the **VRRP** menu.

Set the **VRRP** to **Enabled**.



Add a new VRRP instance.

Click **+ VRRP instance**.

Set the VRRP instance parameters, then click **Create**.

For vCPE-51:

- **Name:** vCPE-5.
- **VRID:** 5.
- **Network interface:** lan.
- **VIP:** 10.20.5.1/24.
- **State:** Master.
- **Priority:** 101.

For vCPE-52:

- **Name:** vCPE-5.
- **VRID:** 5.
- **Network interface:** lan.
- **VIP:** 10.20.5.1/24.
- **State:** Backup.
- **Priority:** 100.

New VRRP instance

Name

VRID ⓘ

Network interface ⓘ

VIP ⓘ

State

vCPE-5

5

lan

10.20.5.1/24

Master

Priority

Advertise interval (sec) ⓘ

☐ Nopreempt ⓘ

101

5

☐ Unicast

Unicast source IP

Unicast peer IP

☐ Authentication

Password

Create

Cancel

Click **Save** in the vCPE-51 and vCPE-52 templates.

4.13.16. Change the address range for the DHCP server and add DHCP option 3 in the vCPE-51 and vCPE-52 templates.

The vCPE-51 and vCPE-52 devices are connected to the same subnet where they function as DHCP servers for workstations. Therefore, it is required to configure a different IP address range in the DHCP server settings to avoid conflicts. It is also required to send a VIP to the workstations as the default gateway.

Open the CPE template, go to the **DHCP** menu, edit the DHCP server on the **lan** interface.

Add **IPv4 gateway** (DHCP option 3), to send a VIP as the default gateway: **10.20.5.1**

In the **vCPE-52** template change the range for vCPE-52, so it won't overlap with vCPE-51 range.

Set:

- **Start IP: 151.**
- **Limit: 50.**

**vCPE-51**

**DHCP**

**BGP**

**VRF**

**OSPF**

**Routing filters**

**PBR**

**BFD**

**Static routes**

**Multicast**

**VRRP**

**CFM**

**+ DHCP server**

Network interface alias	Type	Start IP
lan	Server	10.20.5.51

**DHCP server**

Start IP ⓘ 151

Limit ⓘ 50

DHCP range 10.20.5.151 - 10.20.5.200

Lease time ⓘ 12 hours

**DHCP options**

IPv4 gateway (3) ⓘ 10.20.5.1

DNS servers (6) ⓘ 8.8.8.8

NTP servers (42) ⓘ

**Save** **Cancel**

## 4.14. CPE device registration

### 4.14.1. Add vCPE-3 device.

Connect to the tenant self-service portal (PoC uses the tenant **Demolab**), to do this, click **Connect as Tenant** from the **Tenants** menu or connect to the SD-WAN orchestrator by the administrator of the created tenant.

Go to **SD-WAN → CPE**, then click **+ CPE**.

Specify the vCPE-3 device parameters:

- **Name:** vCPE-3.
- **DPID:** The device DPID is displayed in the command line of the CPE device.
- **State:** Activated.
- **CPE Template:** vCPE-3.
- **Firewall template:** cpe\_firewall\_template.

Click **Next**.

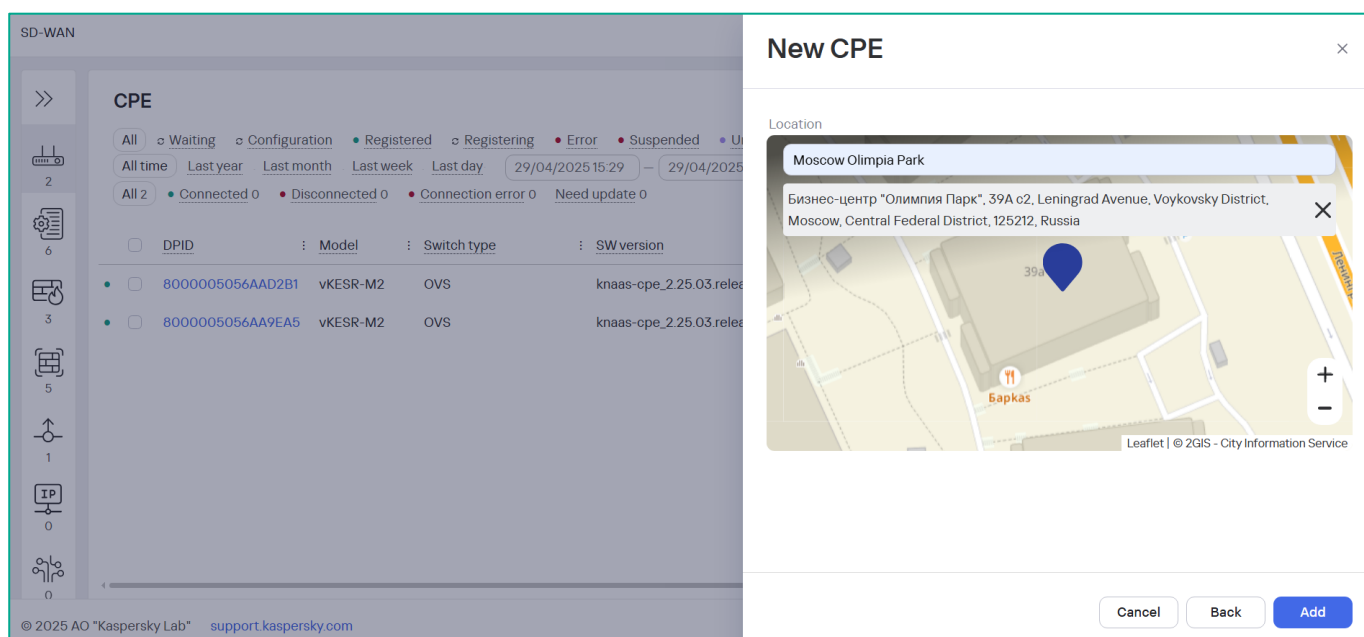
The screenshot displays the Kaspersky SD-WAN interface. On the left, a sidebar shows navigation icons. The main area is titled 'CPE' and contains a table of devices. Overlaid on this is a 'New CPE' configuration window. The window has the following fields and values:

- Name:** vCPE-3
- DPID:** 8000005056AAC4FD
- State:** Enabled (with a help icon)
- Description:** (empty text field)
- UNI template:** (empty dropdown menu)
- CPE template:** vCPE-3
- NetFlow template:** Default NetFlow template (Demolab)
- Firewall template:** cpe\_firewall\_template (Demolab) (Demolab)

At the bottom right of the 'New CPE' window, there are three buttons: 'Cancel', 'Back', and 'Next'. The 'Next' button is highlighted in blue.

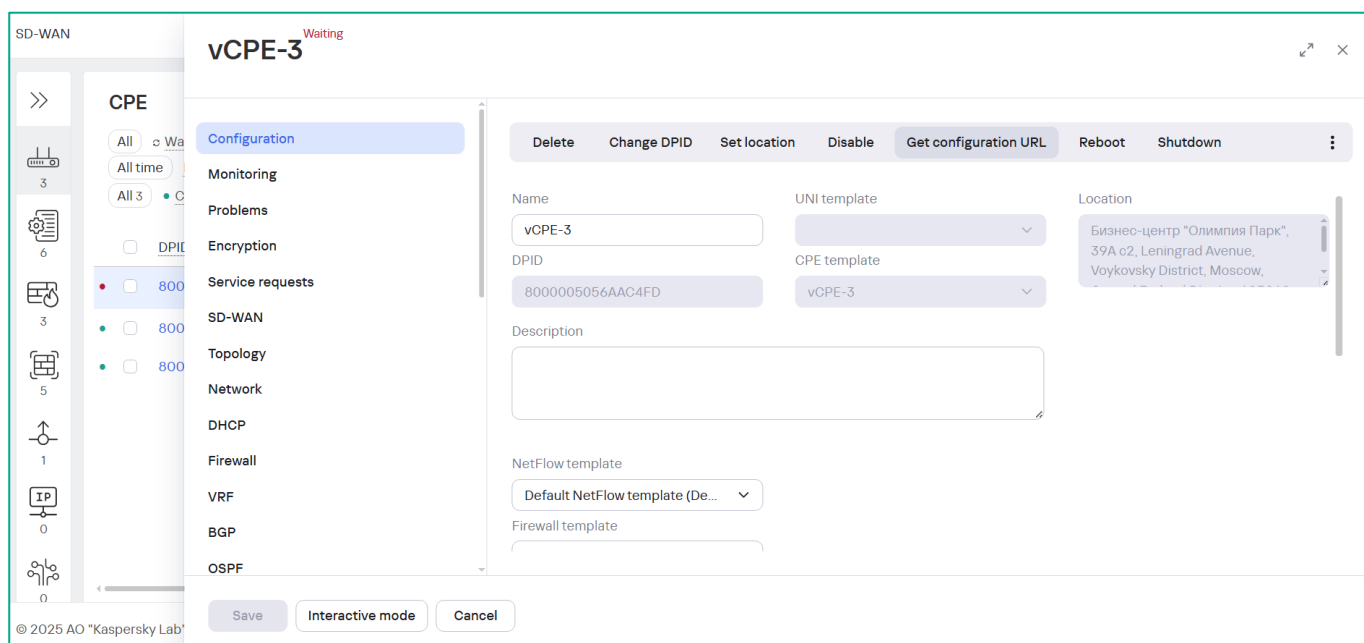


Optionally set the CPE location, then click **Add**.

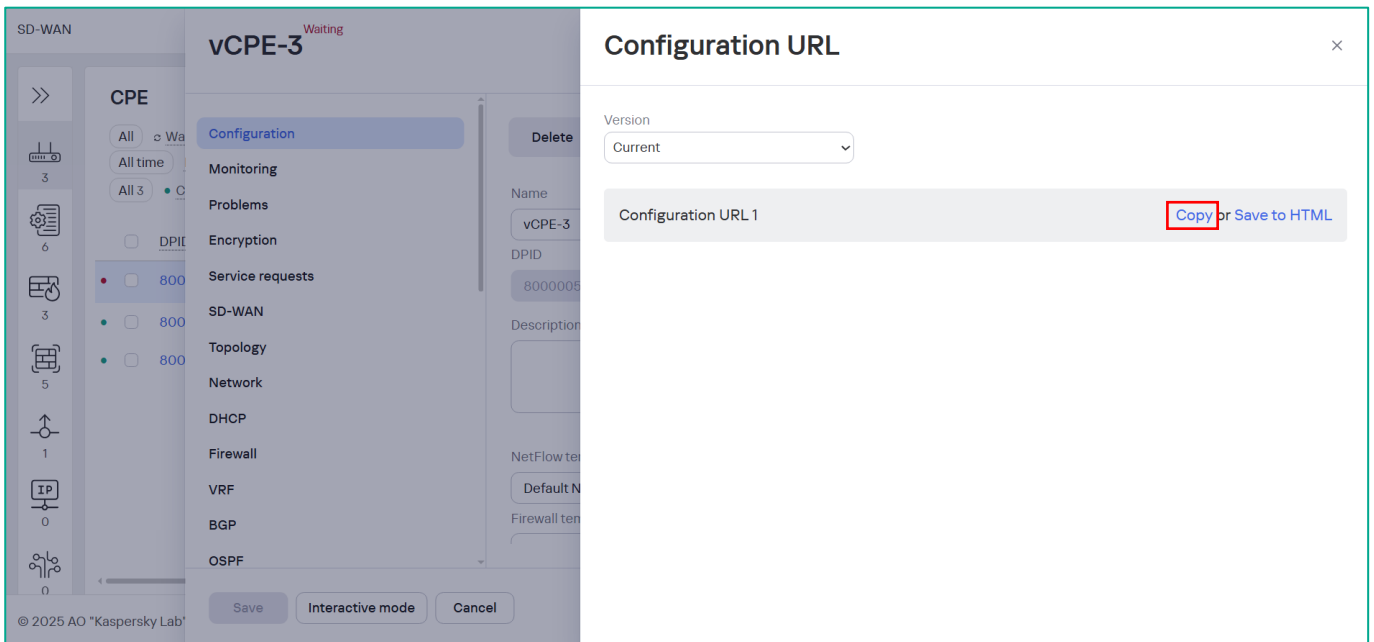


4.14.2. Complete vCPE-3 activation with a Configuration URL.

Go to **SD-WAN → CPE → Configuration**, select vCPE-3 device, then click **Get configuration URL**.



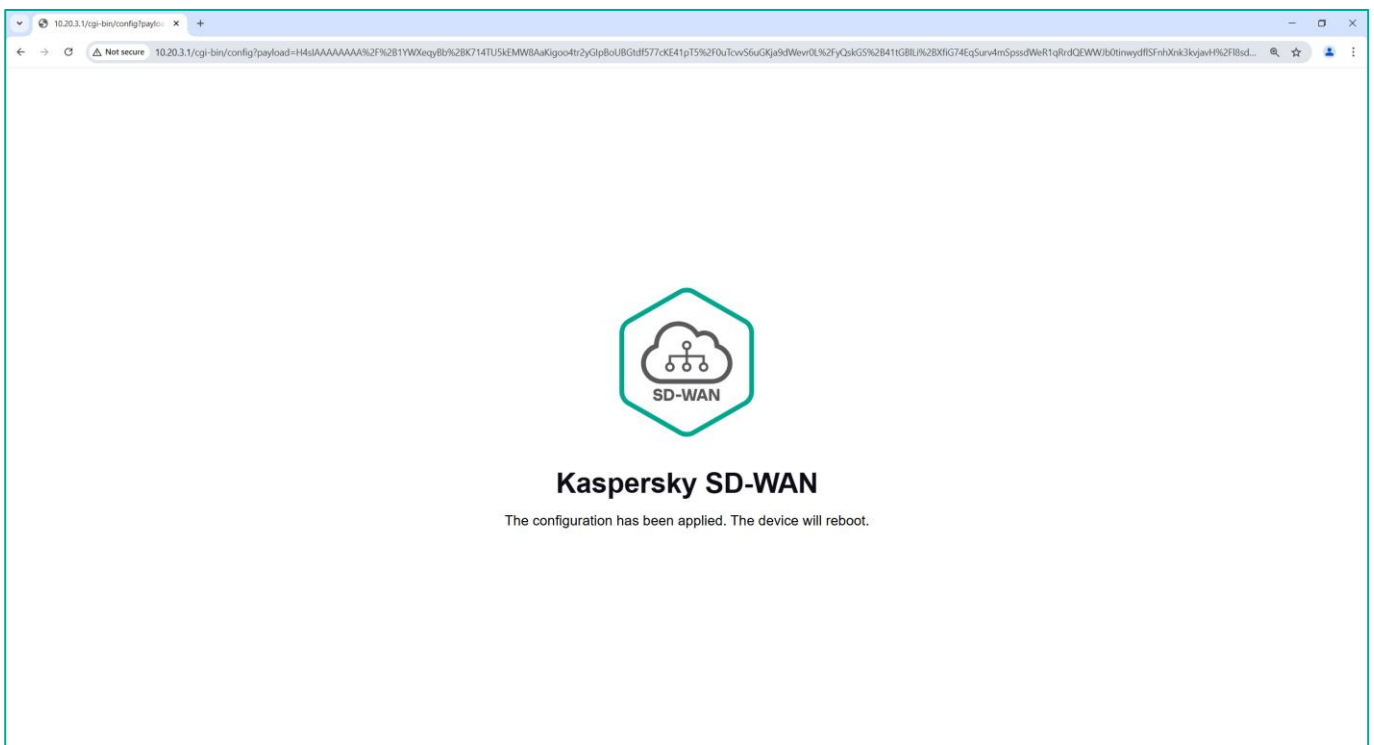
Click **Copy**.



Open the copied link in a browser (you must have network connectivity with the CPE lan interface).

The configuration will be applied to the CPE, and the CPE will automatically reboot.

The configuration URL contains network settings and the CA certificate.



After reboot the CPE device state will change to **Registering**.

**vCPE-3** Registering

Configuration | Monitoring | Problems | Encryption | Service requests | SD-WAN | Topology | Network | DHCP | Firewall | VRF | BGP | OSPF

Buttons: Delete, Show password, Get configuration URL, Reboot, Shutdown, Export SD-WAN settings, Export network interfaces

Name: vCPE-3 | UNI template: [dropdown] | Location: Бизнес-центр "Олимпия Парк", 39А с2, Leningrad Avenue, Vaykovsky District, Moscow.

DPID: 800005056AAC4FD | CPE template: vCPE-3

Description: [text area]

NetFlow template: Default NetFlow template (De...) | Firewall template: cpe\_firewall\_template (Demolab)

Device information table:

Model	SW version	Controller	User	Registered	Update	Management IP	State	Connection
[empty row]								

Buttons: Save, Interactive mode, Cancel

The **Service Request** tab will appear. New service request will be created for CPE registration.

**vCPE-3** Registered

Configuration | Monitoring | Problems | Encryption | **Service requests** | Tags | Scripts

Buttons: Reload service requests, Cancel all service requests

Name	Created	Task ID	Time	Status	Actions
CpeRegistration	29/04/2025 16:39:08	343e395a-9ee5-42b5-86b7-4b07de42170e	2m 17s	Executed	
CpeApplyConfiguration	29/04/2025 16:41:26	d8c09dfb-21b4-4675-a49f-d237e558ff12	0	Executed	

Buttons: Save, Interactive mode, Cancel

Click the **Task ID** for registration details.

Registered

vCPE-3

Configuration

Monitoring

Problems

Encryption

Service requests

Tags

Scripts

SD-WAN

Topology

Network

DHCP

Firewall

VRF

BGP

Save

Interactive mode

CpeRegistration

Created: 29/04/2025 16:39:08

Task ID: 343e395a-9ee5-42b5-86b7-4b07de42170e

Time: 2m 17s

Status: Executed

Name	Status	Time	Attributes
CommutatorAttachCommand	Executed	2m 6s	cluster: SD-WAN Cluster [Demolab: 4f7461d3-0a4b-
CommutatorRenameCommand	Executed	0	name: vCPE-3: 8000005056AAC4FD
CommutatorUpdatePortsStateSet	Executed	0	
CommutatorUpdatePortStateCommand	Executed	0	number: 4800
CommutatorUpdatePortStateCommand	Executed	0	number: 4801
CommutatorSetLinksEncryptionCommand	Executed	0	encrypted: true
CommutatorSetCfmCommand	Executed	0	cfmEnabled: true
CommutatorUpdatePublicPortSettingsSet	Executed	0	
CommutatorUpdatePublicPortSettingsCommand	Executed	0	number: 4800

Refresh

Cancel

vCPE-3 device will change status to **Registered** and **Connected**.

SD-WAN

+ CPE

>>

CPE

Export to CSV...

All

Waiting

Configuration

Registered

Registering

Error

Suspended

Unknown

All time

Last year

Last month

Last week

Last day

29/04/2025 15:29

29/04/2025 15:29

All 3

Connected 0

Disconnected 0

Connection error 0

Need update 0

DPID	Model	Switch type	SW version	Name	Role	Status	State	Connection
8000005056AAC4FD	vKESR-M1	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vCPE-3	CPE	Registered	Enabled	Connected
8000005056AAD2B1	vKESR-M2	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vGW-12	Gateway	Registered	Enabled	Connected
8000005056AA9EA5	vKESR-M2	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vGW-11	Gateway	Registered	Enabled	Connected

vCPE-3 registration is complete.

4.14.3. Complete registration of vCPE-3, vCPE-4, vCPE-51 and vCPE-52 devices.

Repeat steps 4.14 - 4.14.2 for other CPE devices.

Go to **SD-WAN → CPE**. Devices should be **Connected** and in **Registered** status.

CPE									
<div> <span>All</span> <span>Waiting</span> <span>Configuration</span> <span>Registered</span> <span>Registering</span> <span>Error</span> <span>Suspended</span> <span>Unknown</span> </div> <div> <span>All time</span> <span>Last year</span> <span>Last month</span> <span>Last week</span> <span>Last day</span> <span>29/04/2025 15:29</span> - <span>29/04/2025 15:29</span> </div> <div> <span>All 6</span> <span>Connected 0</span> <span>Disconnected 0</span> <span>Connection error 0</span> <span>Need update 0</span> </div>									
<input type="checkbox"/>	DPID	Model	Switch type	SW version	Name	Role	Status	State	Connection
<input checked="" type="checkbox"/>	8000005056AAC6B5	vKESR-M1	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vCPE-52	CPE	Registered	Enabled	Connected
<input checked="" type="checkbox"/>	8000005056AAB512	vKESR-M1	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vCPE-51	CPE	Registered	Enabled	Connected
<input checked="" type="checkbox"/>	8000005056AA35FF	vKESR-M1	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vCPE-4	CPE	Registered	Enabled	Connected
<input checked="" type="checkbox"/>	8000005056AAC4FD	vKESR-M1	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	vCPE-3	CPE	Registered	Enabled	Connected
<input checked="" type="checkbox"/>	8000005056AAD2B1	vKESR-M2	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	VGW-12	Gateway	Registered	Enabled	Connected
<input checked="" type="checkbox"/>	8000005056AA9EA5	vKESR-M2	OVS	knaas-cpe_2.25.03.release.91.bios.amd64	VGW-11	Gateway	Registered	Enabled	Connected

4.14.4. Verify the Management P2M transport service.

Go to **SD-WAN → Infrastructure → SD-WAN cluster → Configuration menu → P2M Services**.

The service should be in the **UP** state.

CPE service interfaces are automatically added with a **Leaf** role.

Infrastructure									
<div> Switches Topology Segments P2P services <b>P2M services</b> M2M services IP multicast services L3 VPN services TAP services Service interfaces Constraints Traffic filters OpenFlow port groups Links </div>									
P2M services									
<div> + Create Search... </div>									
Name	Mode	MAC age (sec)	MAC learn mode	MAC table size	MAC table overload	Endpoints	Status	Description	
SD-WAN management Tunnel	Classic	300	Learn and flood	2000	Flood	St//VGW-11: 8000005056AA9EA5/p.1, Role: Root St//VGW-12: 8000005056AAD2B1/p.1, Role: Root St//vCPE-3: 8000005056AAC4FD/p.1, Role: Leaf St//vCPE-51: 8000005056AAB512/p.1, Role: Leaf St//vCPE-4: 8000005056AA35FF/p.1, Role: Leaf St//vCPE-52: 8000005056AAC6B5/p.1, Role: Leaf	Up		Management

4.14.5. Verify CPE device connection to the controller.

Go to **Infrastructure** → **SD-WAN Cluster** → **Configuration menu** → **Switches**.

CPE devices should be in a **Connected** status.

Infrastructure													
Switches													
+ Add													
Search...													
	Name	Order	Status	Connection	Bloc...	MAC	Interfa...	Pr...	IP	Port	Created	Location	
								se...	address				
<input type="checkbox"/>	vCPE-3: 8000005056AAC4FD	3	Active	Connected	No	00:50:56:aa:c4 fd	sdwan0 sdwan1	Yes No	10.50.5.9 10.50.6.1	34946 35132	29/04/2025 16:41:32	Moscow, Leningrad Avenue, 39A c2	
<input type="checkbox"/>	vCPE-4: 8000005056AA35FF	5	Active	Connected	No	00:50:56:aa:3 5:ff	sdwan1 sdwan0	No Yes	10.50.6.1 6	52378 56942	29/04/2025 16:50:31		
<input type="checkbox"/>	vCPE-51: 8000005056AAB512	4	Active	Connected	No	00:50:56:aa:b 5:12	sdwan0 sdwan1	Yes No	10.50.7.8 10.50.8.1	39176 45890	29/04/2025 16:48:32		
<input type="checkbox"/>	vCPE-52: 8000005056AAC6B5	6	Active	Connected	No	00:50:56:aa:c6 b5	sdwan0 sdwan1	Yes No	10.50.7.9 10.50.8.1	40524 33584	29/04/2025 16:51:47		
<input type="checkbox"/>	vGW-11: 8000005056AA9EA5	1	Active	Connected	No	00:50:56:aa:9e a5	sdwan0	Yes	10.50.1.1 1	45014	29/04/2025 13:28:32	Moscow, Leningrad Avenue, 39A c2	
<input type="checkbox"/>	vGW-12: 8000005056AAD2B1	2	Active	Connected	No	00:50:56:aa:d 2:b1	sdwan0	Yes	10.50.2.1 2	56938	29/04/2025 13:38:31	Moscow, Leningrad Avenue, 39A c2	

4.14.6. Verify the establishment of GENEVE tunnels to the gateways.

Open **Links** menu and check the established GENEVE tunnels to the gateways.

Links (GENEVE tunnels) have been established from all CPEs to each gateway from both SD-WAN interfaces in two directions.

Infrastructure													
Links													
Set default utilization													
Discovered links Active links													
	Source	NAT/Disjoint src	Source IP/port	Destination	NAT/Disjoint dst	Destination							
	vCPE-4: 8000005056AA35FF : 4800	10.50.5.10:4800	N	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N							
	vCPE-4: 8000005056AA35FF : 4801	10.50.6.16:4801	N	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N							
	vCPE-4: 8000005056AA35FF : 4800	10.50.5.10:4800	N	vGW-12: 8000005056AAD2B1 : 4800	10.50.2.12:4800	N							
	vCPE-4: 8000005056AA35FF : 4801	10.50.6.16:4801	N	vGW-12: 8000005056AAD2B1 : 4800	10.50.2.12:4800	N							
	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N	vCPE-4: 8000005056AA35FF : 4800	10.50.5.10:4800	N							
	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N	vCPE-4: 8000005056AA35FF : 4801	10.50.6.16:4801	N							
	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N	vCPE-51: 8000005056AAB512 : 4800	10.50.7.8:4800	N							
	vGW-11: 8000005056AA9EA5 : 4800	10.50.1.11:4800	N	vCPE-51: 8000005056AAB512 : 4801	10.50.8.17:4801	N							

## 5. Traffic management

### 5.1. Creating a Layer 2 Multipoint-to-Multipoint (M2M) transport service

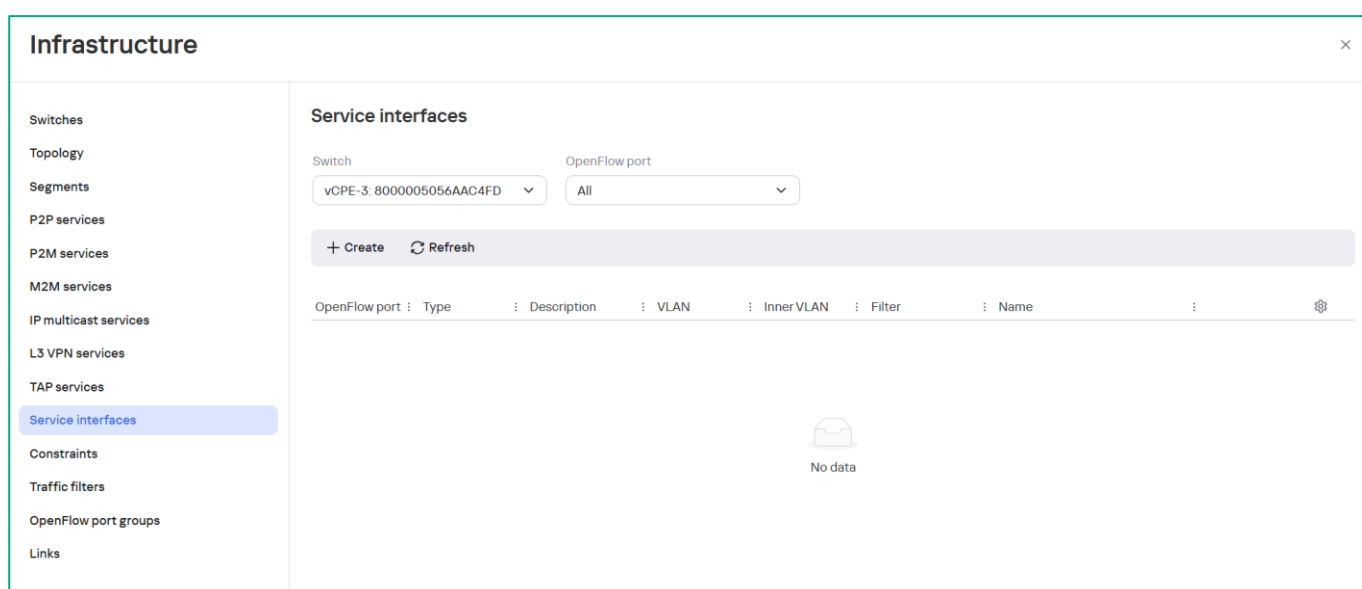
The hub-and-spoke topology is the most commonly used topology within SD-WAN networks. This section describes the hub-and-spoke topology, where remote sites are connected to SD-WAN gateways.

To provide connectivity on top of the established links, an L2 transport service must be created. In this PoC Multipoint-to-Multipoint (E-LAN in the MEF classification) service is used.

M2M is a transport service in which traffic is transmitted between multiple service interfaces without a hierarchy.

#### 5.1.1. Create service interfaces.

Go to **SD-WAN → Infrastructure → SD-WAN controller → Configuration menu → Service Interfaces**.



Select CPE device, then click **+ Create**.

Set service interface parameters:

- **OpenFlow port: 2 (ovs-lan)**. This ovs port is bound to the overlay CPE interface.
- **Type: Access** (both tagged and untagged frames will be accepted on the interface).

Click **Create**.

Repeat for all CPEs and gateways.

5.1.2. Create a L2 M2M transport service.

Go to **Infrastructure → SD-WAN cluster → Configuration menu → M2M Services**.

Click **+ Create**.



Enter the service **Name** and click **Next**.

### New M2M service

Name

L2 M2M

Constraint

Threshold

Balancing mode ⓘ

Per-flow

MAC learn mode

Learn and flood

MAC age (sec)

300

MAC table overload

Flood

MAC table size

100

Description

Cancel

Back

Next

In the **Service endpoints** section, click **+ Add** to the service interfaces created in step 5.1.1.

Set service endpoints parameters:

- **Switch:** select all CPE devices and gateways (in this PoC the M2M transport service is configured for all devices).
- **Service interface:** **Port 2 Access.**
- **QoS:** **Unlimited QoS.**

Click **Next**.

### New M2M service

Service endpoints

+ Add

Switch	Service interface	QoS rule	Inbound filter	Backup switch	Backup service interface
vCPE-3: 8000005056AAC4FD	Port 2, Access	Unlimited-QoS	—	—	—
vCPE-4: 8000005056AA35FF	Port 2, Access	Unlimited-QoS	—	—	—
vCPE-51: 8000005056AAB5...	Port 2, Access	Unlimited-QoS	—	—	—
vCPE-52: 8000005056AAC6...	Port 2, Access	Unlimited-QoS	—	—	—
vGW-11: 8000005056AA9E...	Port 2, Access	Unlimited-QoS	—	—	—
vGW-12: 8000005056AAD2...	Port 2, Access	Unlimited-QoS	—	—	—

Cancel

Back

Next

Keep the default values and click **Create**.

New M2M service

Port groups

+ Add

Cancel

Back

Create

The M2M transport service has been created.

The service is operational and in the **UP** state.

Infrastructure

Switches

Topology

Segments

P2P services

P2M services

M2M services

IP multicast services

L3 VPN services

TAP services

Service interfaces

Constraints

Traffic filters

OpenFlow port groups

Links

M2M services

+ Create

Search...

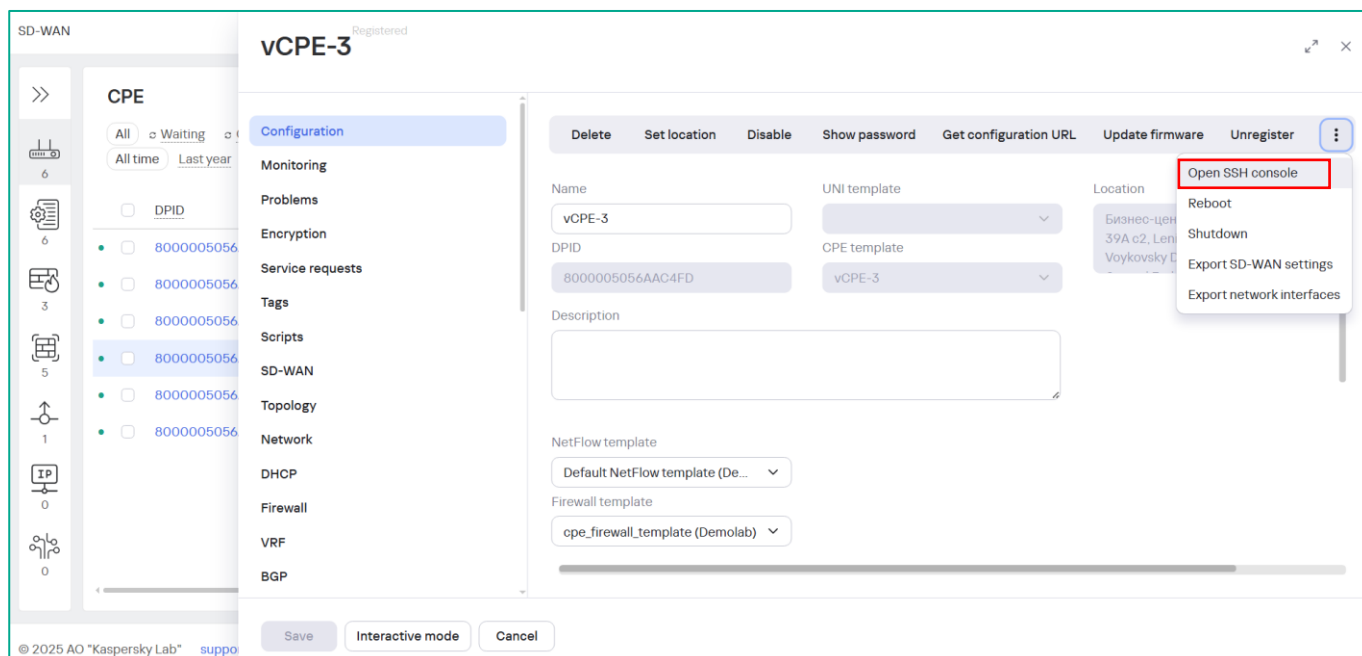
AllUpDownDegraded

Name	MAC age (sec)	MAC learn mode	MAC table size	MAC table overload	Endpoints	Status	Description
L2 M2M	300	Learn and flood	100	Flood	St://vCPE-3: 8000005056AAC4FD/p:2 St://vCPE-4: 8000005056AA35FF/p:2 St://vCPE-51: 8000005056AAB512/p:2 St://vCPE-52: 8000005056AAC6B5/p:2 St://vGW-11: 8000005056AA9EA5/p:2 St://vGW-12: 8000005056AAD2B1/p:2	Up	Management

## 6. Verifying BGP and VRRP protocols on CPEs

### 6.1.1. Verify BGP operation on vCPE-3.

Go to **SD-WAN → CPE → Configuration**, select **vCPE-3** and click **Open SSH Console**. SSH session to the CPE will open in a new browser tab. It is also possible to connect to the CPE device in a separate SSH session.



**Note:** To view the CPE password select device in **SD-WAN → CPE → Configuration** and then click **Show password**. Default user is **root**.

FRRouting is used for BGP. Start FRR shell:

```
vttysh
```

Execute the following commands:

```
show ip route
```

```
show ip bgp sum
```

```
show run
```

The routing table has routes received via BGP from the CPE.

BGP sessions to gateways are established.

```
8000005056AAC4FD# show ip bgp summary

IPv4 Unicast Summary (VRF default):
BGP router identifier 172.16.1.3, local AS number 65500 vrf-id 0
BGP table version 8
RIB entries 13, using 2496 bytes of memory
Peers 2, using 1449 KiB of memory

Neighbor      V      AS  MsgRcvd  MsgSent  TblVer  InQ  OutQ  Up/Down  State/PfxRcd  PfxSnt  Desc
172.16.1.11   4      65500    225     219      0     0     0 00:35:50        6         1 N/A
172.16.1.12   4      65500    225     219      0     0     0 00:35:50        6         1 N/A

Total number of neighbors 2
8000005056AAC4FD#
```

6.1.2. Verify BGP operation on vGW-12.

Connect to the **vGW-12** via the SSH console.

Execute the following command

**ip r**

```
-----
CPEOS knaas-cpe_2.24.09.release.28.amd64, 1730242530
-----

root@8000005056AA9EA5:~# ip r
default via 10.1.4.1 dev eth0 proto static src 10.1.4.11 metric 1
10.0.1.0/24 nhid 91 via 10.1.3.13 dev eth1 proto bgp metric 20
10.1.1.0/24 nhid 91 via 10.1.3.13 dev eth1 proto bgp metric 20
10.1.3.0/24 dev eth1 proto kernel scope link src 10.1.3.11
10.1.4.0/24 dev eth0 proto static scope link metric 100
10.11.13.0/24 dev br-nfvmgmt proto kernel scope link src 10.11.13.73
10.20.3.0/24 nhid 106 via 172.16.1.3 dev overlay proto bgp metric 20
10.20.4.0/24 nhid 108 via 172.16.1.4 dev overlay proto bgp metric 20
10.20.5.0/24 nhid 112 proto bgp metric 20
    nexthop via 172.16.1.52 dev overlay weight 1
    nexthop via 172.16.1.51 dev overlay weight 1
172.16.1.0/24 dev overlay proto kernel scope link src 172.16.1.11
root@8000005056AA9EA5:~#
```

The routing table has routes received via BGP from the CPE.

6.1.3. Verify the connectivity between wst3, wst4, wst5, srv1 and orc1.

**Note:** wst3, wst4 and wst5 obtain IP addresses via DHCP from CPE devices. You will need to renew the IPv4 addresses on workstations after all CPEs have been configured..

Connect to host **wst3** using the SSH console.

Start ping to the IP addresses of hosts **wst4**, **wst5**, **srv1** and **orc1**.

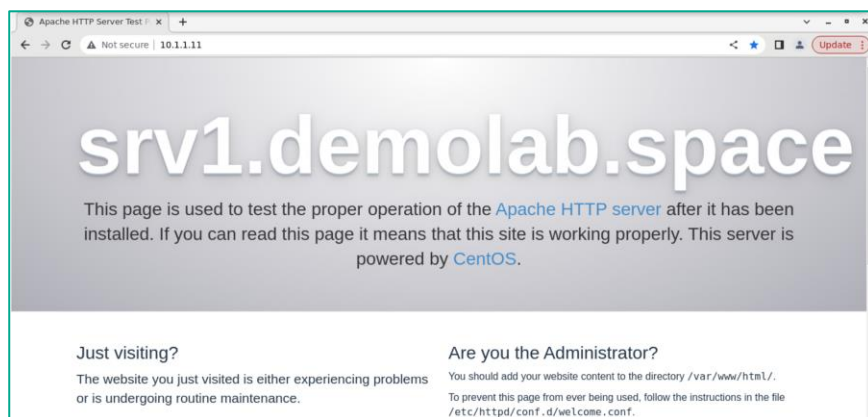
```
[root@wst3 ~]# ping 10.20.4.171
PING 10.20.4.171 (10.20.4.171) 56(84) bytes of data.
64 bytes from 10.20.4.171: icmp_seq=1 ttl=62 time=6.97 ms
64 bytes from 10.20.4.171: icmp_seq=2 ttl=62 time=2.67 ms
^C
--- 10.20.4.171 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.675/4.825/6.975/2.150 ms
[root@wst3 ~]# ping 10.20.5.200
PING 10.20.5.200 (10.20.5.200) 56(84) bytes of data.
64 bytes from 10.20.5.200: icmp_seq=1 ttl=62 time=8.28 ms
64 bytes from 10.20.5.200: icmp_seq=2 ttl=62 time=2.89 ms
^C
--- 10.20.5.200 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.897/5.592/8.288/2.696 ms
[root@wst3 ~]# ping 10.1.1.11
PING 10.1.1.11 (10.1.1.11) 56(84) bytes of data.
64 bytes from 10.1.1.11: icmp_seq=1 ttl=61 time=3.49 ms
```

The ICMP ping should be successful.

Repeat the ping from hosts wst4 and wst5.

6.1.4. Verify the connection to the WWW server on srv1.

Enter **10.1.1.11** (srv1 IP address) in the browser on workstation **wst3**.



The site displays successfully on wst3.

## 6.1.5. Verify VRRP operation on vCPE-51 and vCPE-52.

Connect to the **vCPE-51** and verify that the CPE has assigned a virtual IP address(10.20.5.1) to the **eth2** interface:

```
ip --br a | grep eth2
```

```
-----
CPEOS knaas-cpe_2.24.09.release.28.amd64, 1730242530
-----
root@8000005056AAB512:~# ip -br a | grep eth2
eth2                UP                10.20.5.2/24 10.20.5.1/24 fe80::250:56ff:feaa:dd49/64
root@8000005056AAB512:~#
```

Connect to the **wst5** host and check connectivity with orc1 or external addresses:

```
ping 10.0.1.11
```

```
ping 8.8.8.8
```

```
[root@wst5 ~]# ping 10.0.1.11
PING 10.0.1.11 (10.0.1.11) 56(84) bytes of data.
64 bytes from 10.0.1.11: icmp_seq=1 ttl=61 time=4.08 ms
64 bytes from 10.0.1.11: icmp_seq=2 ttl=61 time=2.13 ms
^C
--- 10.0.1.11 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.136/3.112/4.088/0.976 ms
[root@wst5 ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=111 time=17.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=111 time=13.7 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 13.762/15.698/17.635/1.940 ms
```

Disable the **lan** interface (**eth2**) on **vCPE-51**:

```
ip link set dev eth2 down
```

Connect to the **vCPE-52**.

Verify that the VRRP has assigned a virtual IP address to the **lan** interface (**eth2**) on **vCPE-52**:

```
ip --br a | grep eth2
```

```
-----
CPEOS knaas-cpe_2.24.09.release.28.amd64, 1730242530
-----
root@8000005056AAC6B5:~# ip --br a | grep eth2
eth2                UP                10.20.5.3/24 10.20.5.1/24 fe80::250:56ff:feaa:f104/64
root@8000005056AAC6B5:~#
```

Connect to the **wst5** host and check connectivity with orc1 or external addresses:

```
ping 10.0.1.11
```

```
ping 8.8.8.8
```

```
[root@wst5 ~]# ping 10.0.1.11
PING 10.0.1.11 (10.0.1.11) 56(84) bytes of data.
64 bytes from 10.0.1.11: icmp_seq=1 ttl=61 time=4.08 ms
64 bytes from 10.0.1.11: icmp_seq=2 ttl=61 time=2.13 ms
^C
--- 10.0.1.11 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.136/3.112/4.088/0.976 ms
[root@wst5 ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=111 time=17.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=111 time=13.7 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 13.762/15.698/17.635/1.940 ms
```

Restore the settings after the test is completed.

Enable the **lan(eth2)** interface on **VCPE-51**.

Connect to **VCPE-51** via web SSH console and execute:

```
ip link set dev eth2 up
```

Verify that the virtual IP address has returned on the **lan** interface (**eth2**) of **VCPE-51** (VCPE-51 will become MASTER after a time interval that is determined by the formula: **3 x Advertise interval set skew\_time (256 - Priority) / 256**)). Example: 5 s + (256-101)/256 = 0.605

```
ip --br a | grep eth2
```

```
root@8000005056AAB512:~# ip --br a | grep eth2
eth2                UP                10.20.5.2/24 10.20.5.1/24 fe80::250:56ff:feaa:dd49/64
root@8000005056AAB512:~#
```

## 7. Upgrading the components of the Kaspersky SD-WAN Management System

7.1.1. Prepare the orchestrator host for SD-WAN upgrade.

Upload the **knaas-installer.<release\_name>.gbl.amd64\_en-US\_ru-RU.tar.gz** archive with new container images and installation playbooks to the sdwan user's home directory on the orc1 host.

Use the **/home/sdwan/poc\_aio.yml** inventory file with the installation variables, configured in step 3.2.6.

7.1.2. Start the Kaspersky SD-WAN upgrade.

Start the Kaspersky SD-WAN Management System installation process, which will update the Kaspersky SD-WAN Management System containers. If the container version is the same as the current one, the container will not be changed.

**Note:** The vault password file, created in step 3.2.10, must exist on host orc1!

Set the EULA acceptance parameter:

```
export KNAAS_EULA_AGREED="true"
```

To start updating Kaspersky SD-WAN components, move to the installation playbooks directory:

```
cd knaas-installer.<release_name>.cis.amd64_en-US_ru-RU/
```

Run the installation playbook specifying the path to the vault password file - you will be asked for a privilege escalation password:

**Note:** : You should check the paths to the installation inventory and the vault password files and change them if necessary!

```
ansible-playbook -i inventory/generic -e "@/home/sdwan/poc_aio.yml" -e "@inventory/external/images.yml" -K --vault-password-file \${HOME}/passwords/vault_password.txt knaas/knaas-install.yml
```

After launch, wait for the Kaspersky SD-WAN component installation playbook to finish.

7.1.3. Clear bash history.

```
history -c && history -w
```



## Appendix A. Configuration of infrastructure components

### ISP router

```
!-----  
ISP  
!-----  
!  
vi /etc/sysconfig/network-scripts/ifcfg-ens224  
!  
TYPE=Ethernet  
PROXY_METHOD=none  
BROWSER_ONLY=no  
BOOTPROTO=none  
DEFROUTE=yes  
IPV4_FAILURE_FATAL=no  
IPV6INIT=no  
IPV6_AUTOCONF=no  
IPV6_DEFROUTE=no  
IPV6_FAILURE_FATAL=no  
IPV6_ADDR_GEN_MODE=stable-privacy  
NAME= ens224  
DEVICE=ens224  
ONBOOT=yes  
IPADDR=10.50.1.1  
PREFIX=24  
IPV6_PRIVACY=no  
!  
vi /etc/sysconfig/network-scripts/ifcfg-ens255  
TYPE=Ethernet  
PROXY_METHOD=none  
BROWSER_ONLY=no  
BOOTPROTO=none
```

```
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=no
IPV6_AUTOCONF=no
IPV6_DEFROUTE=no
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens255
DEVICE=ens255
ONBOOT=yes
IPADDR=8.8.8.8
PREFIX=24
IPV6_PRIVACY=no
!
nano /etc/sysconfig/network
NOZEROCONF=yes
!
nano -w /etc/resolv.conf
nameserver 8.8.8.8
!
sysconfig restart network
!
#Upgrade packages:
yum update -y
yum install -y epel-release
yum update -y
!
#Install and configure NTP:
yum install ntp ntpdate -y
timedatectl set-ntp true
ntpdate ntp.demolab.space
ntpdate -d ntp.demolab.space
timedatectl status
```

```
!  
nano -w /etc/chrony.conf  
chronyc tracking  
chronyc sourcestats  
!  
nano /etc/sysctl.conf  
net.ipv4.ip_forward=1  
!  
sysctl -w net.ipv4.ip_forward=1  
!  
#Disable SELINUX :  
sed -i s/^SELINUX=.*$/SELINUX=disabled/ /etc/selinux/config  
setenforce 0  
!  
#Disable Firewall:  
systemctl disable firewalld  
systemctl stop firewalld  
systemctl disable NetworkManager  
systemctl stop NetworkManager  
systemctl enable network  
systemctl start network  
!  
#Install iptables:  
yum install iptables-services -y  
systemctl enable iptables  
systemctl start iptables  
!  
#Drop iptables :  
iptables -P INPUT ACCEPT  
iptables -P FORWARD ACCEPT  
iptables -P OUTPUT ACCEPT  
iptables -t nat -F  
iptables -t mangle -F
```

```

iptables -F
iptables -X
!
iptables -A INPUT -i ens192 -j ACCEPT
iptables -A INPUT -i ens224 -j ACCEPT
iptables -A INPUT -i ens256 -j ACCEPT
iptables -A INPUT -i ens225 -j ACCEPT
iptables -A INPUT -i ens257 -j ACCEPT
iptables -A INPUT -i ens162 -j ACCEPT
iptables -A INPUT -i ens194 -j ACCEPT
!
iptables -A FORWARD -i ens192 -j ACCEPT
iptables -A FORWARD -i ens224 -j ACCEPT
iptables -A FORWARD -i ens256 -j ACCEPT
iptables -A FORWARD -i ens225 -j ACCEPT
iptables -A FORWARD -i ens257 -j ACCEPT
iptables -A FORWARD -i ens162 -j ACCEPT
iptables -A FORWARD -i ens194 -j ACCEPT
!
#Save iptables settings:
service iptables save
!
#Display iptables settings:
iptables -L -n -v!
yum install nano net-tools bind-utils tcpdump traceroute -y
!
DHCP
!
yum install dhcp
nano -w /etc/dhcp/dhcpd.conf
!
subnet 10.50.5.0 netmask 255.255.255.0 {
    range 10.50.5.3 10.50.5.253;

```

```
option domain-name-servers 8.8.8.8;  
option domain-name "demolab.space";  
option routers 10.50.5.1;  
option broadcast-address 10.50.5.255;  
default-lease-time 600;  
max-lease-time 7200;  
}
```

```
subnet 10.50.6.0 netmask 255.255.255.0 {  
    range 10.50.6.3 10.50.6.253;  
    option domain-name-servers 8.8.8.8;  
    option domain-name "demolab.space";  
    option routers 10.50.6.1;  
    option broadcast-address 10.50.6.255;  
    default-lease-time 600;  
    max-lease-time 7200;  
}
```

```
subnet 10.50.7.0 netmask 255.255.255.0 {  
    range 10.50.7.3 10.50.7.253;  
    option domain-name-servers 8.8.8.8;  
    option domain-name "demolab.space";  
    option routers 10.50.7.1;  
    option broadcast-address 10.50.7.255;  
    default-lease-time 600;  
    max-lease-time 7200;  
}
```

```
subnet 10.50.8.0 netmask 255.255.255.0 {  
    range 10.50.8.3 10.50.8.253;  
    option domain-name-servers 8.8.8.8;  
    option domain-name "demolab.space";  
    option routers 10.50.8.1;
```

```
option broadcast-address 10.50.8.255;
default-lease-time 600;
max-lease-time 7200;
}
!
#Check DHCP configuration:
dhcpd -t -cf /etc/dhcp/dhcpd.conf
!
#Enable dhcpd:
systemctl enable dhcpd
#Star DHCPD:
systemctl start dhcpd
!
#DHCP must be enabled to specific interfaces.
nano -w /etc/sysconfig/dhcpd
#Add interfaces:
DHCPDARGS=ens225,ens257,ens162,ens194
!
#Restart dhcpd:
systemctl restart dhcpd
!restart dhcpd
cat /var/lib/dhcpd/dhcpd.leases
!
```

## R13 router

**Note:** If you change the mgmt subnet in step 4.1.5, you need to change the gateway IP addresses in the routes to the new ones.

**#Drop iptables:**

```
iptables -P INPUT ACCEPT
```

```
iptables -P FORWARD ACCEPT
```

```
iptables -P OUTPUT ACCEPT
```

```
iptables -t nat -F
```

```
iptables -t mangle -F
```

```
iptables -F
```

```
iptables -X
```

```
!
```

```
iptables -A FORWARD -j ACCEPT
```

```
iptables -A INPUT -j ACCEPT
```

```
!
```

```
service iptables save
```

```
iptables -L -n -v
```

```
!
```

```
# yum install -y https://github.com/FRRouting/frr/releases/download/frr-5.0.1/frr-5.0.1-2018070501.el7.centos.x86_64.rpm
```

```
# nano -w /etc/frr/daemons
```

```
!
```

```
zebra=yes
```

```
bgpd=yes
```

```
1
```

```
# systemctl enable frr && systemctl start frr
```

```
# systemctl status frr
```

```
# vtysh
```

```
!
```

```
conf t
```

```
!
```

```
router bgp 65613
```

```
bgp router-id 10.1.3.13
```

```
bgp log-neighbor-changes
timers bgp 10 30
neighbor 10.1.3.11 remote-as 65500
neighbor 10.1.3.12 remote-as 65500
address-family ipv4 unicast
redistribute connected
exit-address-family
!
end
write
!
```



## R14 router

**Note:** If you change IP plan in 2.4, you must use the new host IP address orc1 and the public IP address (replace 10.0.1.11 and 10.50.1.14).

```
#Drop iptables:

iptables -P INPUT ACCEPT

iptables -P FORWARD ACCEPT

iptables -P OUTPUT ACCEPT

iptables -t nat -F

iptables -t mangle -F

iptables -F

iptables -X

!

# Create an iptables rule to allow forwarding between the internal (ens224) and
# external (ens192) interfaces:

iptables -A FORWARD -i ens224 -o ens192 -j ACCEPT

!

#Allow to forward packers only for established sessions.

#

iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT

!

# SNAT configuration

iptables -t nat -A POSTROUTING -s 10.1.1.0/24 -o ens192 -j SNAT --to-source 10.50.1.14

iptables -t nat -A POSTROUTING -s 10.1.3.0/24 -o ens192 -j SNAT --to-source 10.50.1.14

iptables -t nat -A POSTROUTING -s 10.0.1.0/24 -o ens192 -j SNAT --to-source 10.50.1.14

!

# DNAT configuration

iptables -A FORWARD -i ens192 -o ens224 -j ACCEPT

iptables -t nat -A PREROUTING -p tcp --dport 443 -i ens192 -j DNAT --to-destination 10.0.1.11

iptables -t nat -A PREROUTING -p tcp --dport 6653 -i ens192 -j DNAT --to-destination 10.0.1.11

iptables -t nat -A PREROUTING -p tcp --dport 6654 -i ens192 -j DNAT --to-destination 10.0.1.11

iptables -t nat -A PREROUTING -p tcp --dport 6655 -i ens192 -j DNAT --to-destination 10.0.1.11

iptables -t nat -A PREROUTING -p tcp --dport 6656 -i ens192 -j DNAT --to-destination 10.0.1.11
```

```
iptables -t nat -A PREROUTING -p tcp --dport 10051 -i ens192 -j DNAT --to-destination 10.0.1.11
!  
service iptables save  
iptables -L -n -v  
iptables -t nat -L -n -v
```

## R11 router

```
#Drop iptables:
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
iptables -t nat -F
iptables -t mangle -F
iptables -F
iptables -X
!

# Create an iptables rule to allow forwarding between the internal (ens224)
# and external (ens192) interfaces:
iptables -A FORWARD -i ens224 -o ens192 -j ACCEPT
!

# Allow to forward packers only for established sessions.
iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
!

# SNAT configuration:
iptables -t nat -A POSTROUTING -s 10.1.4.0/24 -o ens192 -j SNAT --to-source 10.50.1.11
!

# DNAT configuration:
iptables -A FORWARD -i ens192 -o ens224 -j ACCEPT
iptables -t nat -A PREROUTING -p udp --dport 4800 -i ens192 -j DNAT --to-destination 10.1.4.11
!

service iptables save
iptables -L -n -v
iptables -t nat -L -n -v
```

## R12 router

```
#Drop iptables:
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
iptables -t nat -F
iptables -t mangle -F
iptables -F
iptables -X

# Create an iptables rule to allow forwarding between the internal (ens224)
# and external (ens192) interfaces:
iptables -A FORWARD -i ens224 -o ens192 -j ACCEPT

# Allow to forward packets only for established sessions.
iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT

# SNAT configuration:
iptables -t nat -A POSTROUTING -s 10.1.5.0/24 -o ens192 -j SNAT --to-source 10.50.2.12

# DNAT configuration:
iptables -A FORWARD -i ens192 -o ens224 -j ACCEPT
iptables -t nat -A PREROUTING -p udp --dport 4800 -i ens192 -j DNAT --to-destination 10.1.5.12

service iptables save

iptables -L -n -v
iptables -t nat -L -n -v
```

## Appendix B. Acceptance test plan

**Note:** All tests must be performed sequentially.

N	Test	PoC item	Description	Expected result	Actual result, comments
<b>Basic orchestrator setup</b>					
1	Authentication in the orchestrator web interface	3.3	<ol style="list-style-type: none"> <li>1. Open the URL of the orchestrator's web interface in a browser.</li> <li>2. Enter the system administrator credentials and click <b>Login</b>.</li> </ol>	Authentication is successful. The Dashboard section opens.	
2	Change the user password	3.3.2	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Users</b> menu.</li> <li>2. Select <b>Administrator</b> user.</li> <li>3. Click <b>Management → Change password</b>, enter new password and click <b>Save</b>.</li> </ol>	The password for the admin user has been successfully changed.	
3	Verification of the created domain.	4.1.1	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Infrastructure</b> menu.</li> <li>2. Select <b>Domain</b> in the <b>Resources</b> section.</li> </ol>	The domain is displayed in the resource list.	
4	Verification of the created data center.	4.1.2	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Infrastructure</b> menu.</li> <li>2. Select <b>Data center</b> in the <b>Resources</b> section.</li> </ol>	The data center is displayed in the resource list.	
5	Connection to the Zabbix monitoring system	4.1.3	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>System → Monitoring</b> menu.</li> <li>2. Click <b>Test connection</b>.</li> </ol>	The connection was successfully configured and the connection test was successful.	
6	Verification of the Zabbix Proxy settings.	4.1.4	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Infrastructure → System resources</b> menu.</li> <li>2. Check Zabbix proxy settings.</li> </ol>	The configured Zabbix proxy settings are displayed in the orchestrator interface.	

N	Test	PoC item	Description	Expected result	Actual result, comments
7	Verify the created IP address pool for the management network.	4.1.5	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Infrastructure</b> → <b>IPAM</b> menu.</li> <li>2. Verify the management network pool created during PoC execution.</li> </ol>	The added IP address pool is displayed in the subnet list.	
8	Verification of the PNF controller descriptor settings.	4.1.7-4.1.9	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Catalog</b> menu.</li> <li>2. In the <b>Physical network functions</b> section, find the added PNF controller descriptor</li> <li>3. Check that the parameters in the tabs match the settings made during PoC execution.</li> </ol>	The controller descriptor was added successfully.	
9	Verification of the SD-WAN service template.	4.3	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Catalog</b> menu.</li> <li>2. In the <b>Templates</b> section, find the created SD-WAN service template.</li> <li>3. Verify that the parameters in the tabs match the settings made during PoC.</li> </ol>	The SD-WAN service template is created and added to Orchestrator.	
10	Verification of the created Tenant.	4.4.1	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Tenants</b> menu.</li> <li>2. Check tenants.</li> </ol>	The created tenant is successfully displayed in the <b>Tenants</b> section.	
11	Adding a user to tenant.	4.4.4	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to <b>Tenants</b> menu.</li> <li>2. Select the tenant created during the PoC.</li> <li>3. Verify the added users in the <b>Users</b> section of the tenant.</li> </ol>	The user has been successfully added to the tenant and is displayed in the <b>Users</b> section of the tenant.	
12	Verification of the deployed SD-WAN service.	4.4.5	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN</b> → <b>SD-WAN instances</b> menu.</li> <li>2. Find the SD-WAN service deployed during the PoC execution.</li> </ol>	SD-WAN service deployment has been deployed: a green indicator is displayed in the orchestrator's web interface for the deployed service.	
13	Verification of the CA certificate for CPE devices.	4.7	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN</b> → <b>Certificates</b> menu.</li> <li>2. View the uploaded certificates.</li> </ol>	The added CA certificate is displayed in the orchestrator's web interface.	

N	Test	PoC item	Description	Expected result	Actual result, comments
<b>CPE devices operations</b>					
14	Verification of SD-WAN gateway templates	4.6	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE templates</b> menu.</li> <li>2. Check SD-WAN gateways templates.</li> </ol>	1. The SD-WAN gateway templates are displayed in the orchestrator with configurations corresponding to the PoC.	
15	Verification of CPE devices templates.	4.13	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE templates</b> menu.</li> <li>2. Check SD-WAN CPE devices templates.</li> </ol>	1. The SD-WAN CPE devices templates are displayed in the orchestrator with configurations corresponding to the PoC.	
16	Verification of firewall zones.	4.5.1, 4.12.1	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → Firewall zones</b> menu.</li> <li>2. Check firewall zones.</li> </ol>	1. The added zones are displayed in the orchestrator	
17	Verification of firewall templates.	4.5, 4.12	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → Firewall templates</b> menu.</li> <li>2. Check firewall templates.</li> </ol>	1. The firewall templates are displayed in the orchestrator with configurations corresponding to the PoC.	
18	Verification of SD-WAN gateway connectivity.	4.9-4.9.4	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE</b> menu.</li> <li>2. Check SD-WAN gateways state.</li> </ol>	1. SD-WAN gateways have been successfully added to the orchestrator and are in the <b>Registered</b> state..	
19	Verification of the management transport service.	4.10	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>Infrastructure → SD-WAN Cluster → Configuration menu → P2M services</b>.</li> <li>2. In the displayed menu check <b>SD-WAN managementTunnel</b> service.</li> </ol>	1. <b>SD-WAN managementTunnel</b> services is in <b>UP</b> state.	

N	Test	PoC item	Description	Expected result	Actual result, comments
20	Verification of CPE device connectivity.	4.14	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE</b> menu.</li> <li>2. Check CPE devices state.</li> </ol>	1. CPE devices have been successfully added to the orchestrator and are in the <b>Registered</b> state..	
21	Access to the CPE CLI console from the orchestrator web interface.	4.10.3	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE</b> menu.</li> <li>2. Select <b>vcPE-3</b>.</li> <li>3. In the <b>Configuration</b> menu click <b>Open SSH console</b></li> <li>4. Repeat for all CPE devices and SD-WAN gateways.</li> </ol>	1. The CPE CLI console opens successfully from the Orchestrator web interface.	
22	Verification of the monitoring operation.	4.9.7	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to the <b>SD-WAN → CPE</b> menu.</li> <li>2. Select <b>vcPE-3</b>.</li> <li>3. Go to Monitoring and sequentially select the data to display in the selector.</li> <li>4. Repeat for all CPE devices and SD-WAN gateways.</li> </ol>	1. The monitoring subsystem is running, and the Orchestrator web interface is successfully displaying statistics for CPEs.	
23	Verification of automatic link creation between gateways and CPE devices.	4.9.5, 4.14.6	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to <b>Infrastructure → SD-WAN Cluster → Configuration menu → Links</b> menu.</li> <li>2. View the displayed list of established links.</li> </ol>	1. GENEVE tunnels have been successfully established between SD-WAN gateways and CPEs.	



N	Test	PoC item	Description	Expected result	Actual result, comments
<b>Transport services operation</b>					
24	Verification of the created service interfaces.	5.1.1	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to <b>Infrastructure</b> → <b>SD-WAN cluster</b> → <b>Configuration menu</b> → <b>Service Interfaces</b>.</li> <li>2. Select <b>vCPE-3</b> and port <b>2 (ovs-lan)</b> to view CPEs service interfaces. Find service interfaces with <b>access</b> type.</li> <li>3. Repeat for all CPE devices and SD-WAN gateways.</li> </ol>	1. Service interfaces for CPE and SD-WAN gateway devices were successfully created.	
25	Verification of the M2M transport service.	5.1.2	<ol style="list-style-type: none"> <li>1. In the orchestrator web interface go to <b>Infrastructure</b> → <b>SD-WAN cluster</b> → <b>Configuration menu</b> → <b>M2M services</b>.</li> <li>2. In the displayed menu, find the M2M transport service.</li> </ol>	1. M2M transport service has been successfully created and is in the <b>UP</b> state.	

kaspersky

kaspersky

[www.kaspersky.com/](http://www.kaspersky.com/)  
[www.securelist.com](http://www.securelist.com)

© 2025 AO Kaspersky Lab.  
All rights reserved. Registered trademarks and service marks are the property of their respective owners