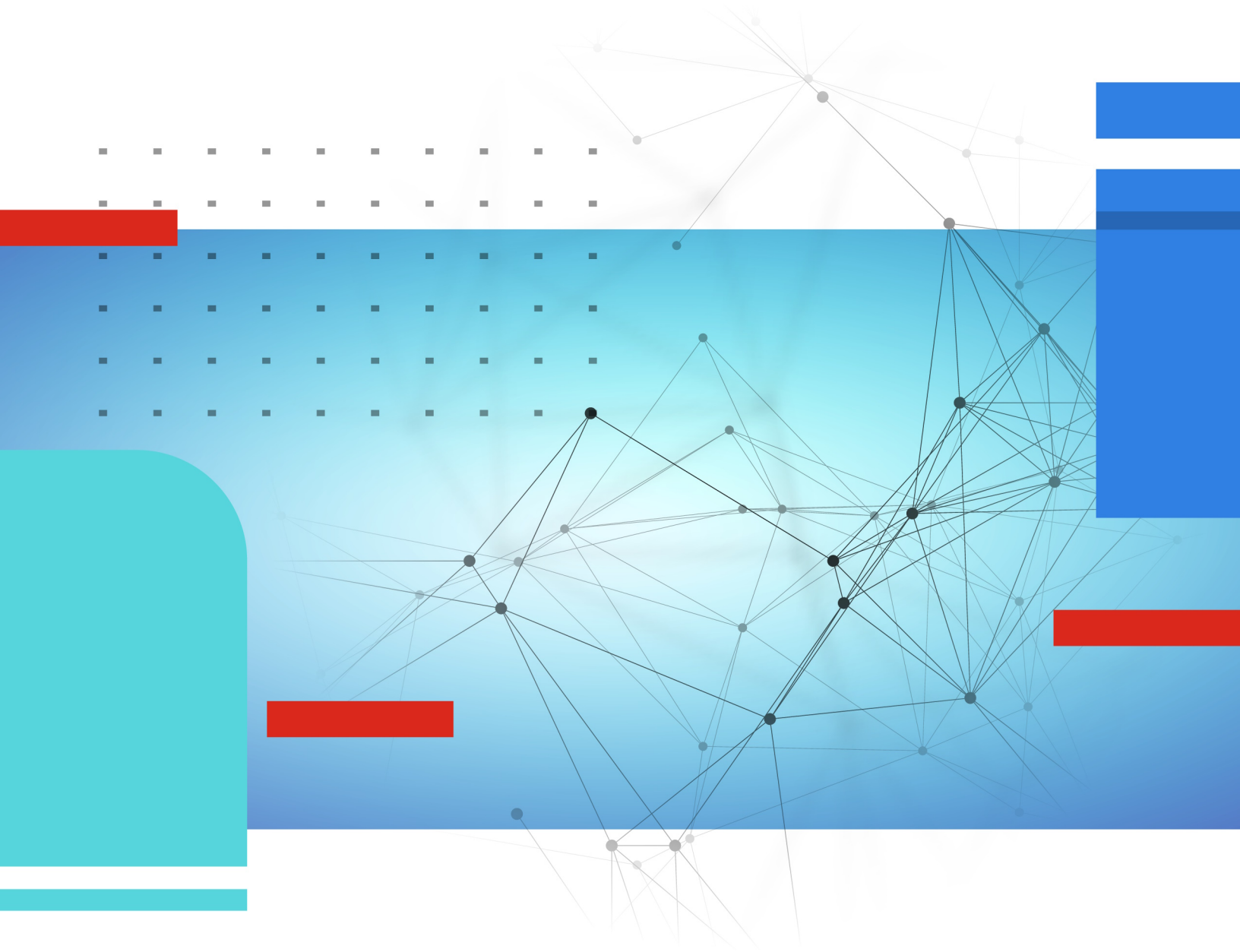
KVM Deployment Guide

FortiProxy 7.6



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

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

FORTINET BLOG
https://blog.fortinet.com

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

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

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

FORTIGUARD LABS
https://www.fortiguard.com

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

FEEDBACK
Email: techdoc@fortinet.com

August 14, 2025
FortiProxy 7.6 KVM Deployment Guide
45-760-1090707-20250814

TABLE OF CONTENTS

Change log

Date	Change Description
2024-10-22	Initial document release.
2025-08-14	Updated Getting started on page 5.

Getting started

FortiProxy is a secure web proxy that protects employees against internet-borne attacks by incorporating multiple detection techniques such as web filtering, DNS filtering, data loss prevention, antivirus, intrusion prevention, and advanced threat protection. It helps enterprises enforce internet compliance using granular application control. High-performance physical and virtual appliances deploy on-site to serve small, medium, and large enterprises

FortiProxy provides multiple detection methods such as reputation lookup, signature-based detection, and sandboxing to protect against known malware, emerging threats, and zero-day malware. It also intercepts outgoing client connections to the internet and has some firewall capabilities. However, the primary focus of FortiProxy is to be a secure web gateway solution that provides visibility, compliance, web security, and threat protection for any organization.

This document describes how to deploy a FortiProxy-VM in a KVM environment. More information about configuring and using FortiProxy is available in the Fortinet Document Library.

In the initial setup, the following ports are used:

- DNS lookup — UDP 53
- FortiGuard licensing — TCP 443

Evaluation license

FortiProxy-VM can be evaluated with a free 15-day trial license that includes most features, except:

- HA
- FortiGuard updates
- Technical support

You do not need to manually upload the trial license; it is built-in. The trial period begins the first time you start FortiProxy-VM. When the trial expires, most functionality is disabled. You must purchase a license to continue using FortiProxy-VM.

License sizes

VM licenses are available in the following sizes:

	Evaluation	VM02	VM04	VM08	VM16	VMUL
Maximum number of CPUs	2	4	8	16	32	Unlimited

	Evaluation	VM02	VM04	VM08	VM16	VMUL
Memory (GB)	2	Unlimited				
Number of disks (boot + storage)	1+1	1+2	1+2	1+4	1+8	16 total

The maximum number of IP sessions varies by license and by available vRAM, just as it does for hardware models. For more information, see the FortiProxy Datasheet.

License validation

FortiProxy-VM must periodically revalidate its license with the Fortinet Distribution Network (FDN). If it cannot contact the FDN for 30 days, access to the FortiProxy-VM web UI and CLI are locked.

By default, FortiProxy-VM attempts to contact FDN over the internet. If the management port cannot access the internet (for example, in closed network environments), it is possible for FortiProxy-VM to validate its license with a FortiManager that has been deployed on the local network to act as a local FDS (FortiGuard Distribution Server).

On the FortiProxy-VM, specify the FortiManager IP address for the “override server” in the FortiGuard configuration:

```
config system central-management
    set type fortimanager
    config server-list
        edit 1
            set server-type update
            set server-address <FortiManager IP address for updates>
        next
        edit 2
            set server-type rating
            set server-address <FortiManager IP address for web filter ratings>
        next
    end
    set include-default-servers disable
end
```

TCP port 8890 is the port where the built-in FDS feature listens for requests. For more information on the FortiManager local FDS feature, see the FortiManager Administration Guide. Although FortiManager can provide FortiGuard security service updates to some Fortinet devices, for FortiProxy, its FDN features can provide license validation only.

Preparing for deployment

This documentation assumes that before deploying the FortiProxy-VM on the KVM virtual platform, you have addressed the following requirements:

Virtual environment

For best performance, install FortiProxy-VM on a “bare metal” hypervisor. Hypervisors that are installed as applications on top of a general-purpose operating system (Windows, Mac OS X, or Linux) host and have fewer computing resources available due to the host OS’s own overhead.

Hardware-assisted virtualization (VT) must be enabled in the BIOS.

You have installed the KVM software on a physical server with sufficient resources to support the FortiProxy-VM and all other VMs deployed on the platform.

If you configure the FortiProxy-VM to operate in transparent mode, or include it in an high availability (HA) cluster, configure any virtual switches to support the FortiProxy-VM's operation before you create the FortiProxy-VM.

VM Environment	Tested Versions
KVM	RHEL 7.1/Ubuntu 12.04 and later CentOS 6.4 (qemu 0.12.1) and later

Management software

You can access the VM using a virtual machine manager, such as virt-manager.

Connectivity

The FortiProxy-VM requires an internet connection to contact FortiGuard to validate its license.

Registering the FortiProxy-VM

When you purchase a FortiProxy-VM, you receive an email that contains a registration number. This registration number is used to download the software, your purchased license, and also to register your purchase with Fortinet Customer Service & Support so that your FortiProxy-VM will be able to validate its license with Fortinet.

Many Fortinet customer services such as firmware updates, technical support, and FortiGuard services require product registration.

For registration instructions, see Registering products in the FortiCloud Account ServicesAsset Management guide.

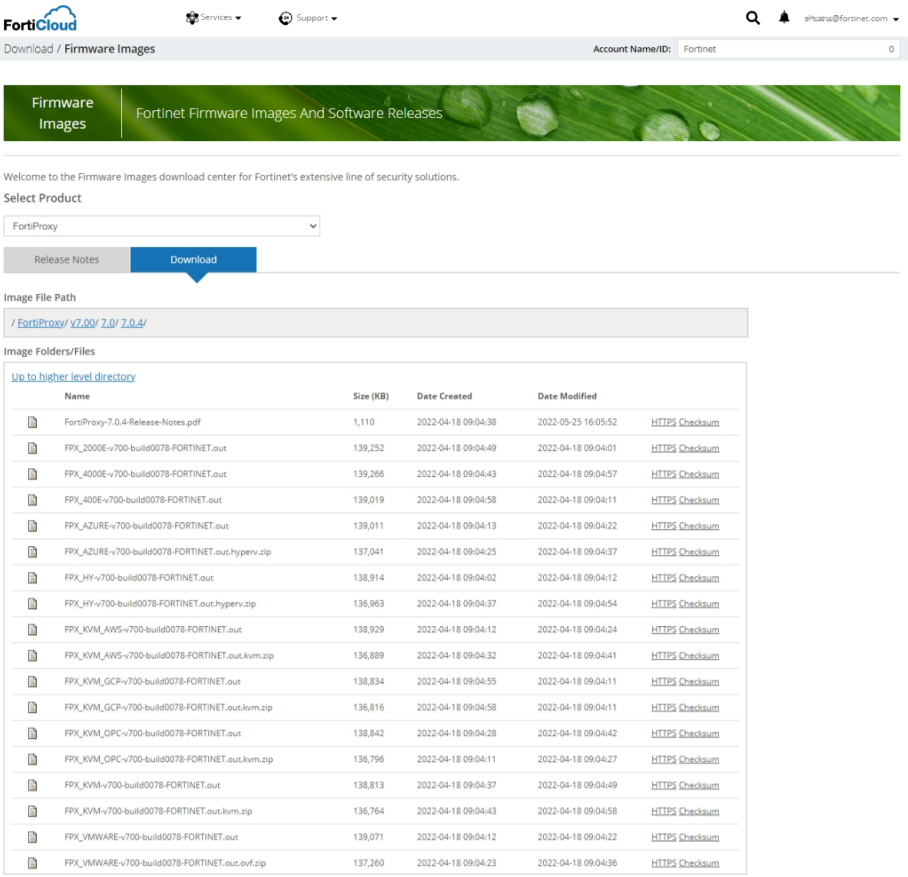
For information about downloaded the license file, see Viewing licenses and keys in the Product details topic of the FortiCloud Account ServicesAsset Management guide.

Downloading the FortiProxy-VM deployment package

FortiProxy-VM deployment packages can be downloaded from the Customer Service & Support.

To download the VM deployment package:

1. Log in to your FortiCloud account.
2. Go to *Support* > *Firmware Download*.
3. In the *Select Product* list, select *FortiProxy*.
4. Select the *Download* tab.
5. Browse to the appropriate directory for the version that you need to download.



6. Download the firmware `.zip` file by clicking the *HTTPS* link to its right.

   The `.out` image files are for upgrades of existing installations only and cannot be used for a new installation.
7. Extract the .zip file contents to a folder.

Deployment package contents

The *FPX_KVM-vxxx-buildxxxx-FORTINET.out.kvm.zip* file contains the *fortiproxy.qcow2* file that is used for the installation. You must manually create a log disk and specify the virtual hardware settings.

Deploying FortiProxy-VM

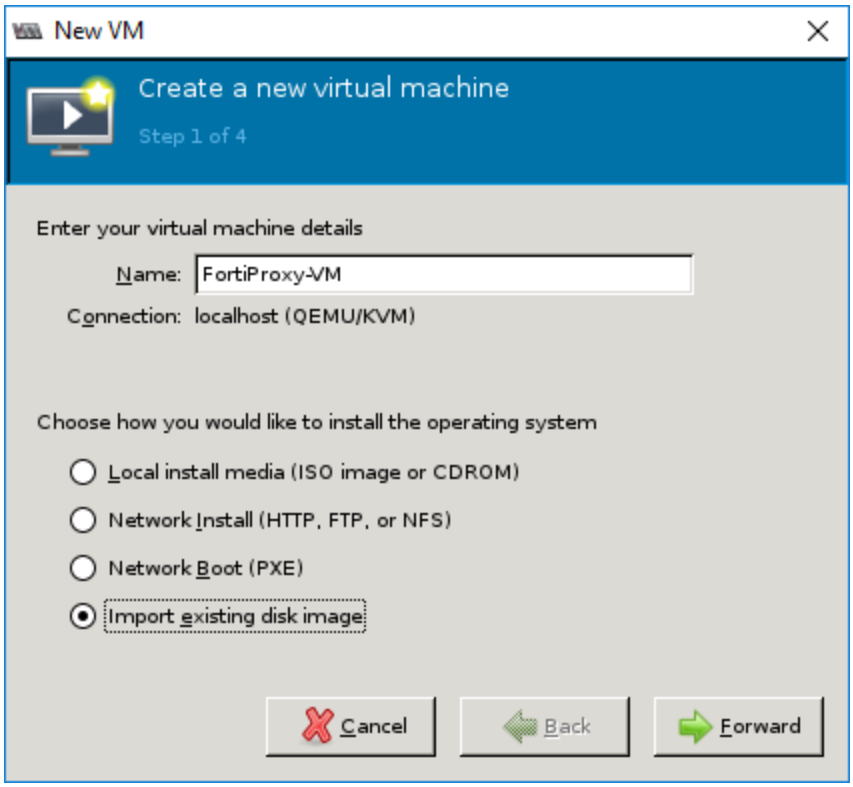
After you have downloaded the FPX_KVM-vxxx-buildxxxx-FORTINET.out.kvm.zip file and extracted the package contents to a folder on your server, you can deploy the FortiProxy-VM on kernel-based virtual machines (KVM) by importing a disk image.

This chapter covers the following topics:

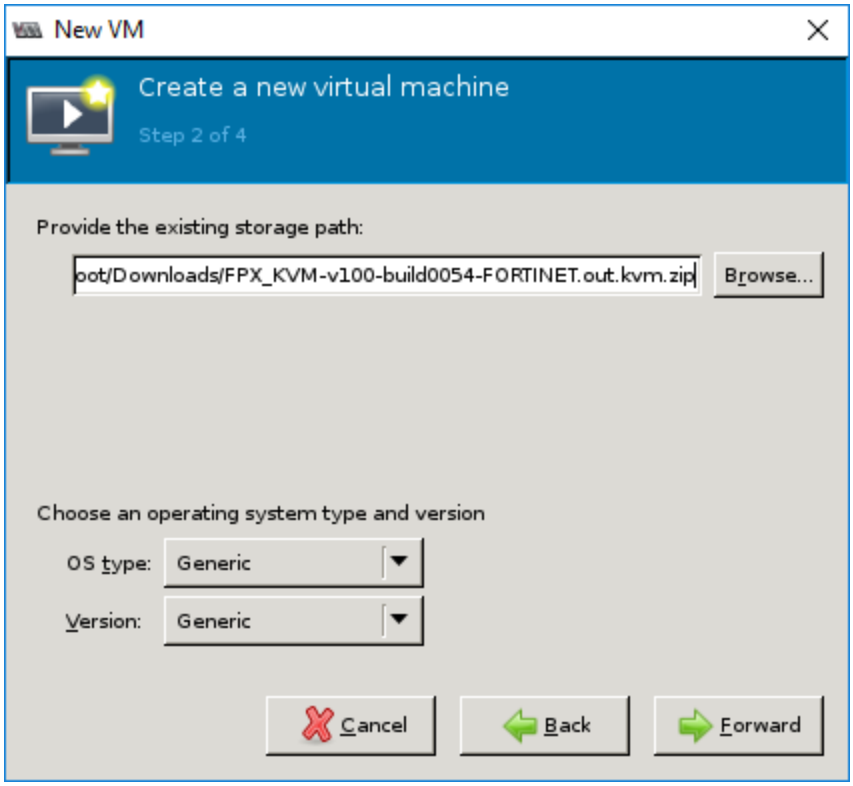
Import the FortiProxy-VM and configure its hardware settings

To import the FortiProxy-VM:

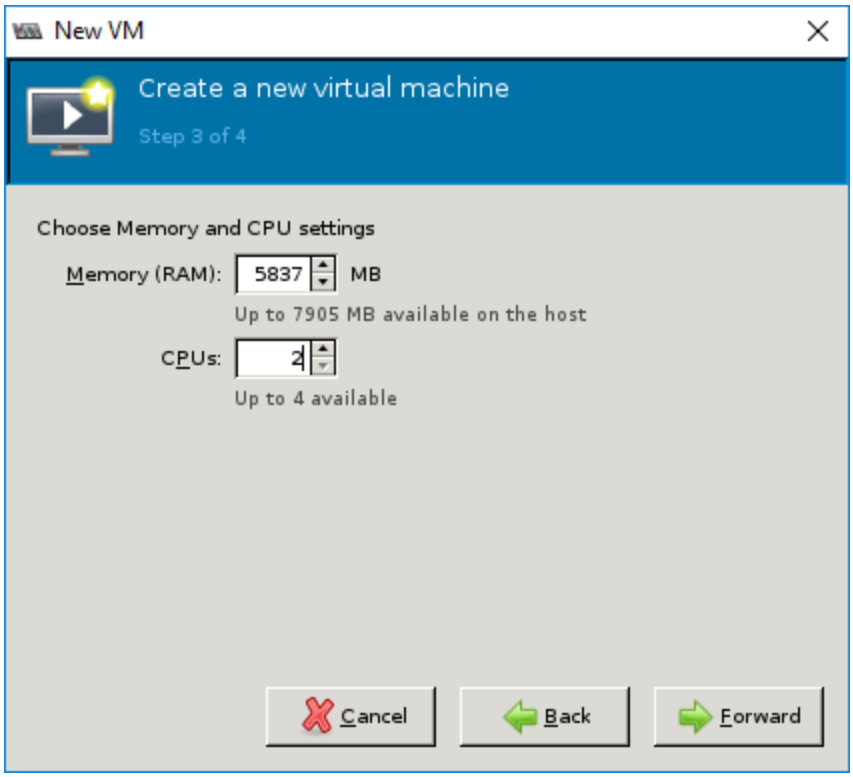
1. On the KVM host server, launch Virtual Machine Manager (virt-manager) and then select *New* to create a new virtual machine.
2. Enter a name for the virtual machine (for example, `FortiProxy-VM`).
3. Select *Import existing disk image*, then click *Forward*.



4. Click *Browse*, navigate to the `FPX_KVM-vxxx-buildxxxx-FORTINET.out.kvm.zip` file, and select it.
5. Use the default values for *OS Type* and *Version*, then click *Forward*.

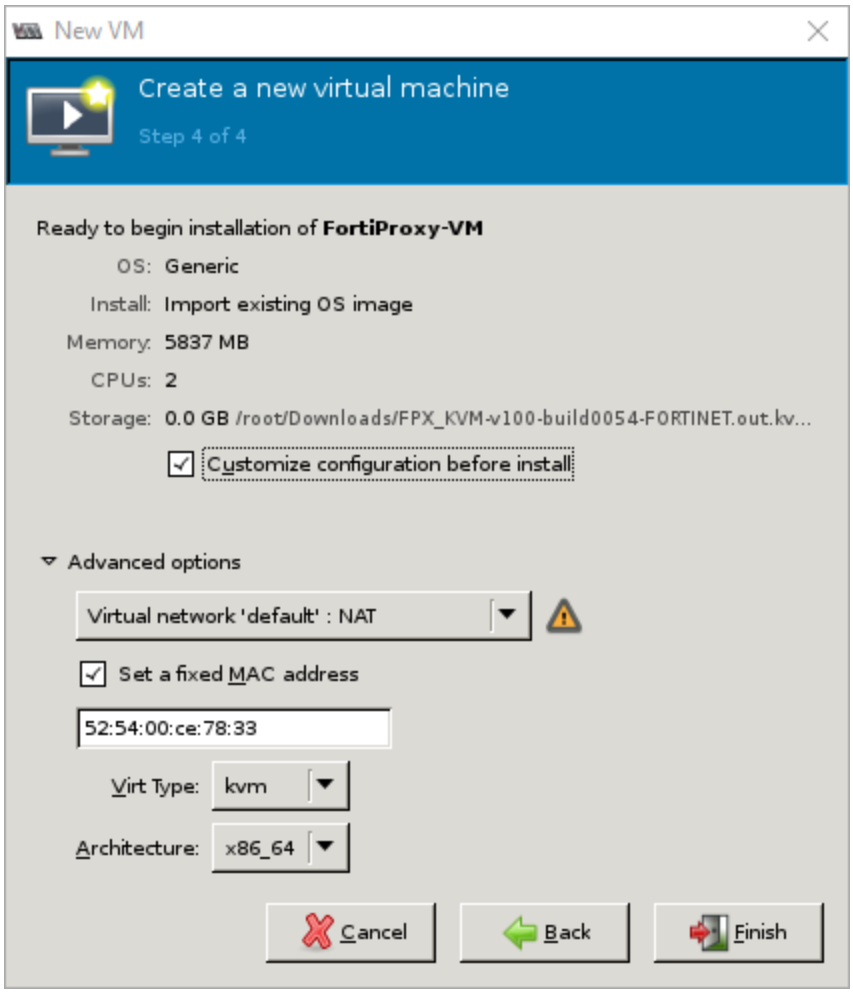


6. Specify the amount of memory and number of CPUs to allocate to this virtual machine. Ensure that the values do not exceed the maximums for your license.

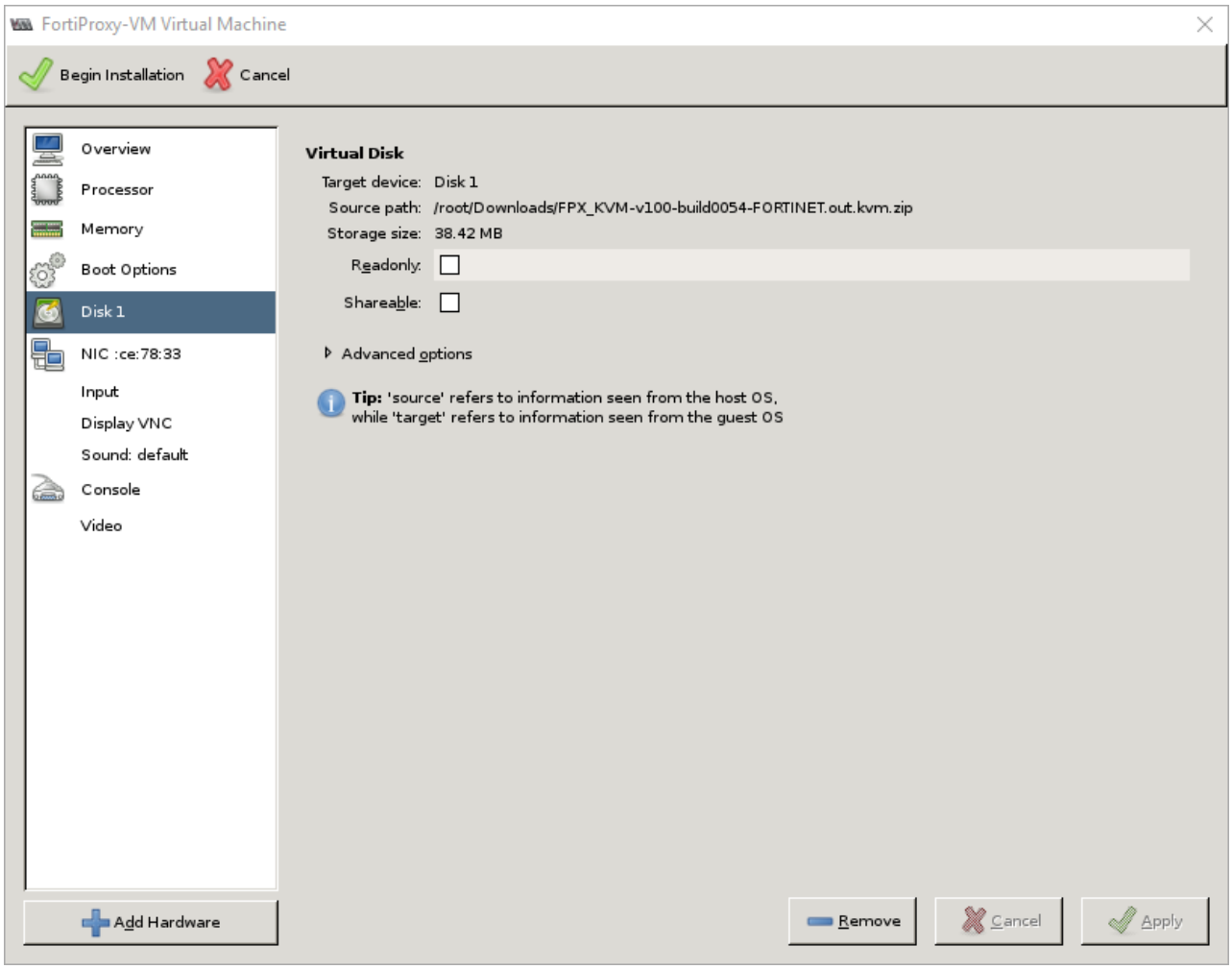


Fortinet recommends that you use at least 4 GB of memory.

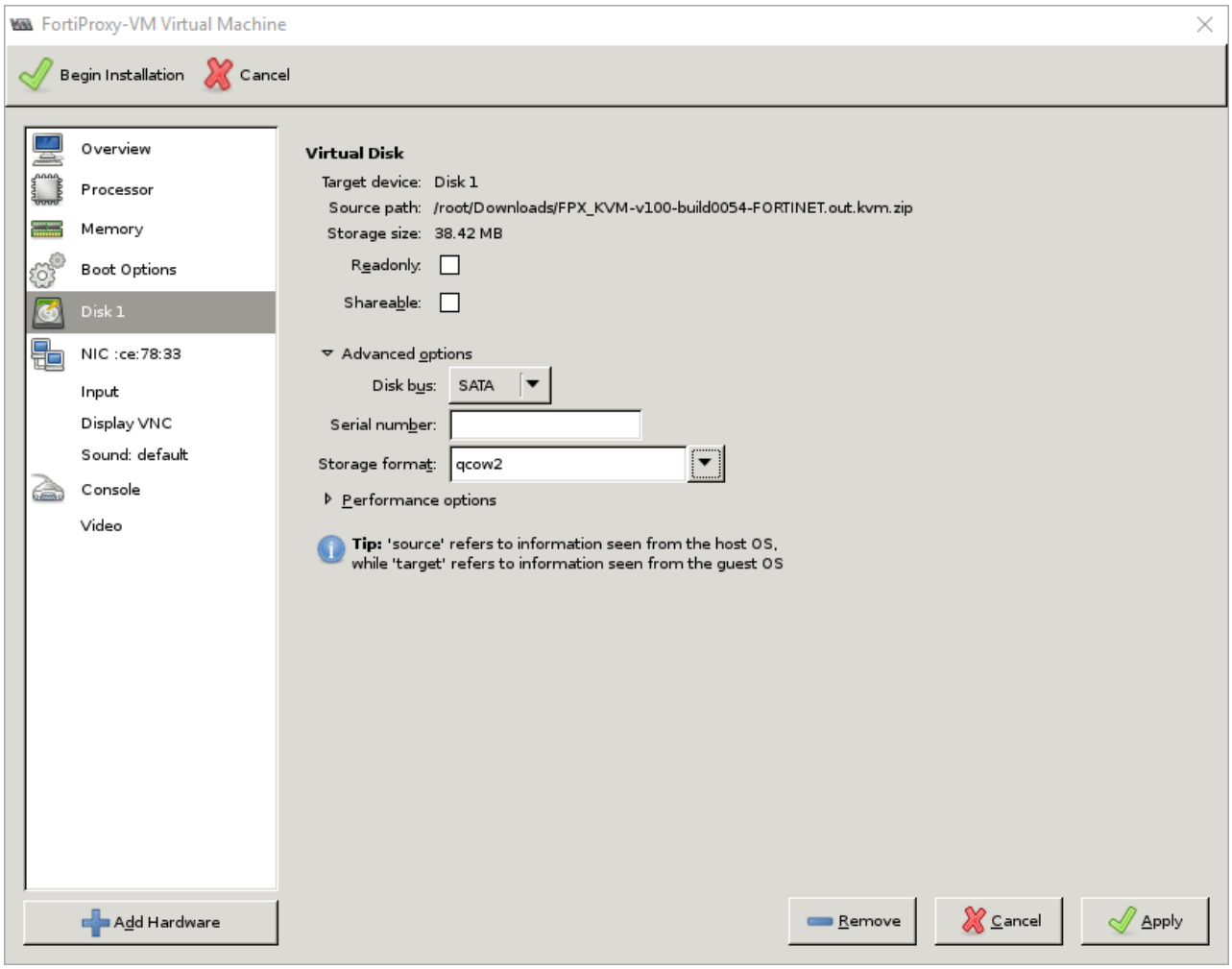
7. Click *Forward*.
8. Select *Customize configuration before install*, then click *Finish*.



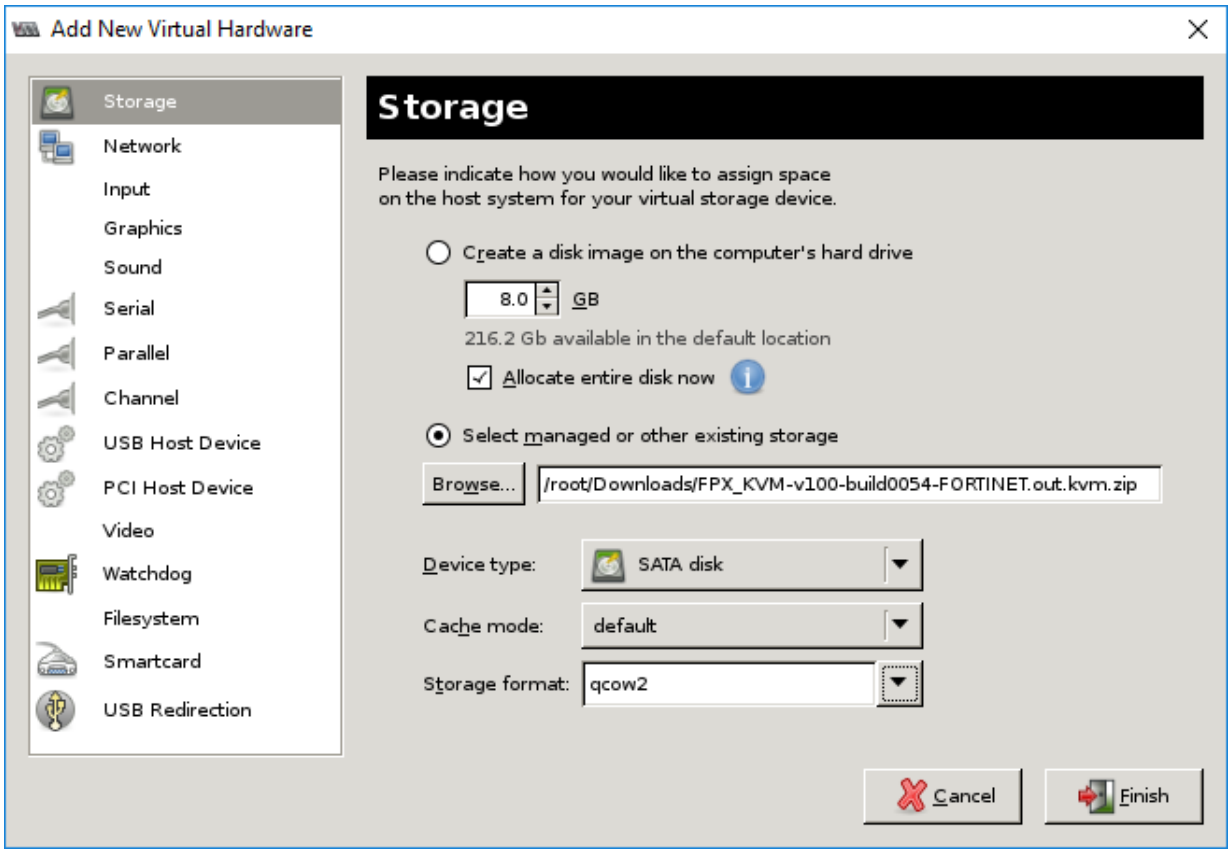
9. Select *Disk 1* to display its properties.



10. Under *Advanced options*, set *Disk bus* to *SATA* and set *Storage format* to *qcow2*.



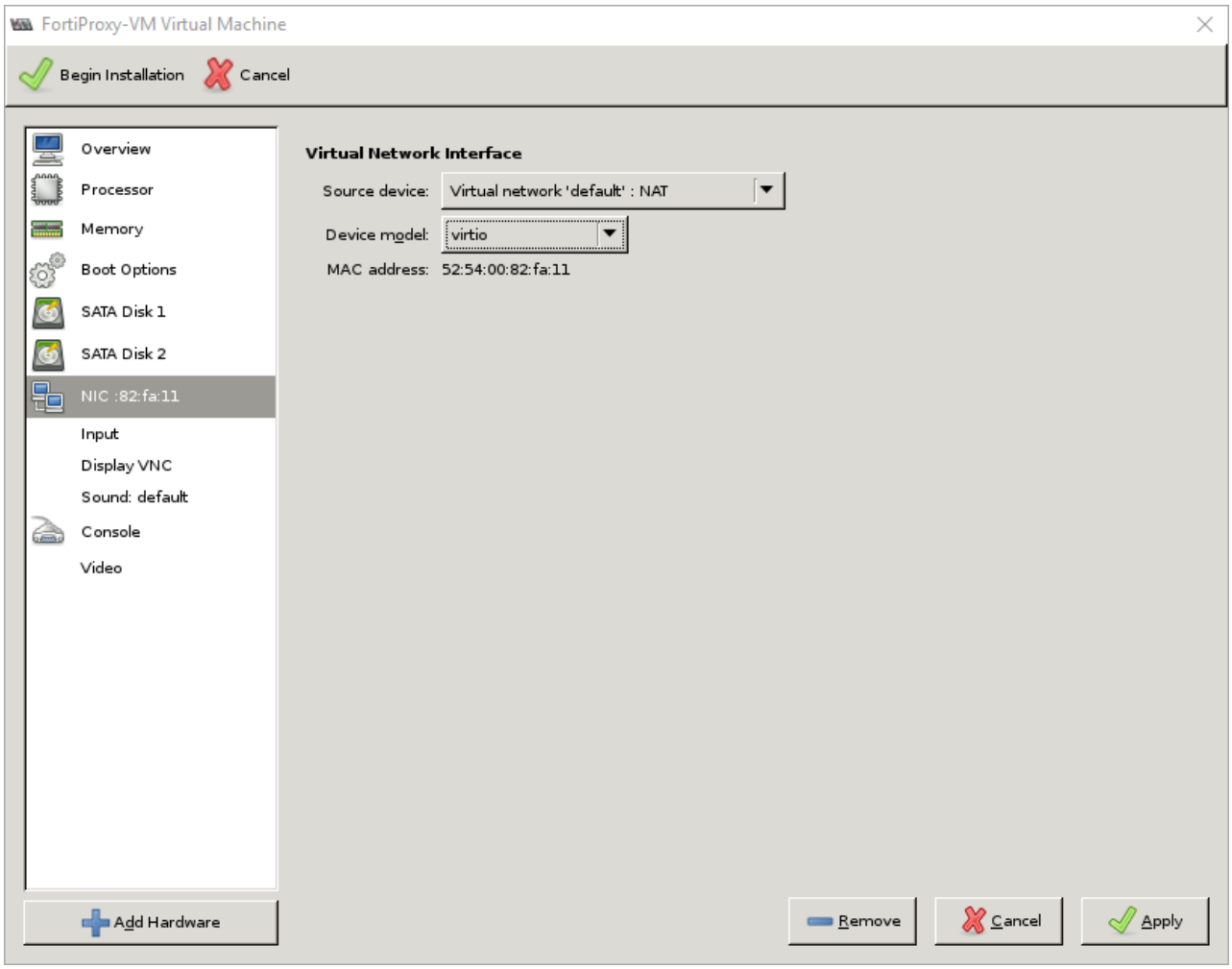
11. Click *Apply*.
12. Click *Add Hardware* to add a new virtual storage device:
    a. Select *Storage* in the left pane.
    b. Select *Select managed or other existing storage*.
    c. Click *Browse*, navigate to `FPX_KVM-vxxx-buildxxxx-FORTINET.out.kvm.zip`, and select it.
    d. Set *Device type* to *SATA disk*.
    e. Set *Storage format* to *qcow2*.



f. Click *Finish*.

13. Select *NIC* to display its properties.

14. Set *Device model* to *virtio*, then click *Apply*.

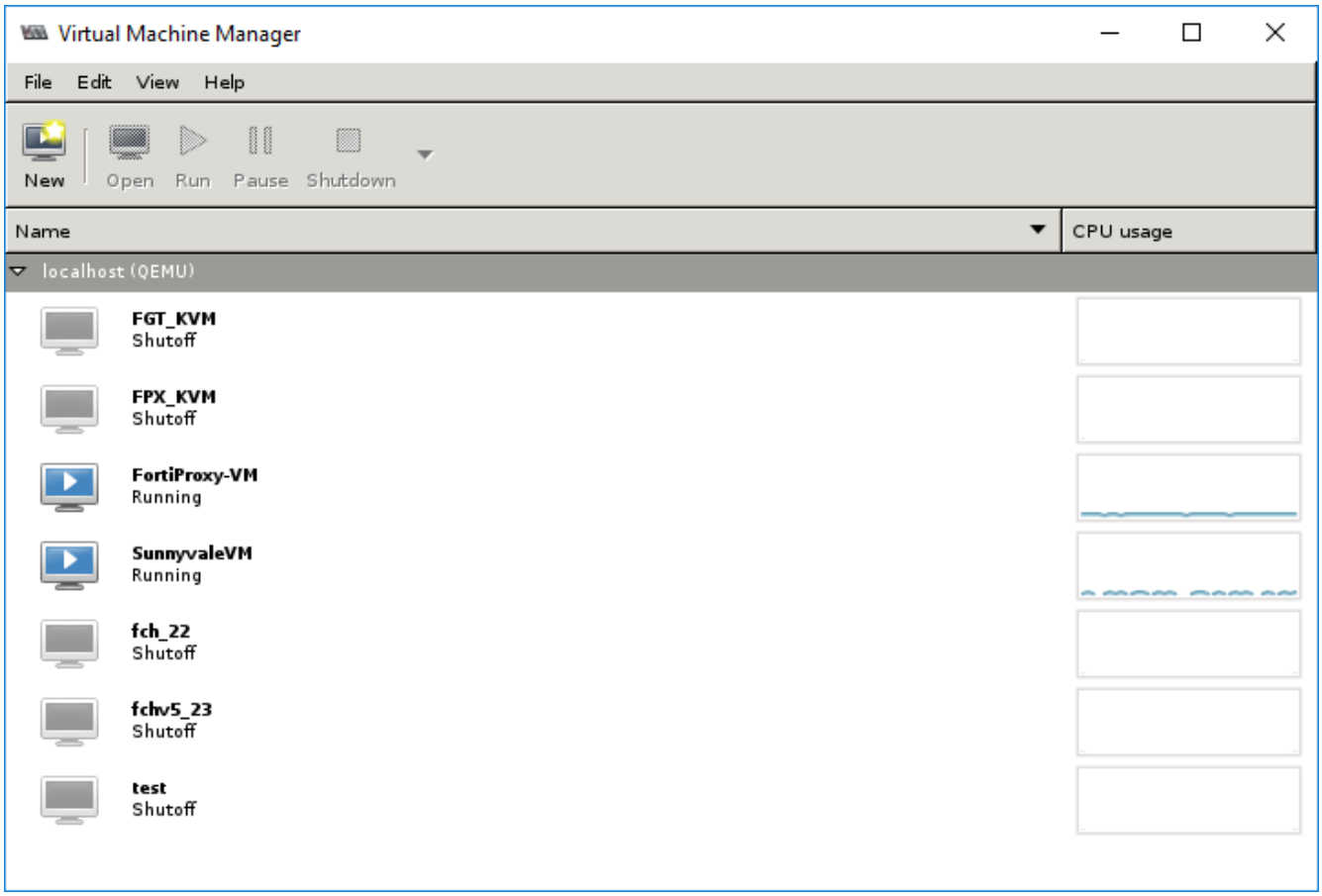


15. Click *Begin Installation*.

Start the FortiProxy-VM

You can now power on your FortiProxy-VM.

On the KVM host server, launch Virtual Machine Manager (virt-manager), then select the virtual appliance and click *Run*.



After the VM starts, proceed with the Initial settings on page 19

Initial settings

The first time that you start the FortiProxy-VM, you will only have access through the console window of your KVM environment. After you configure one FortiProxy network interface with an IP address and administrative access, you can access the FortiProxy-VM GUI.

Every FortiProxy-VM includes a 15-day trial license. During this time the VM operates in evaluation mode. Before using the VM, you must upload the license file that you downloaded from Customer Service & Support upon registration.

More information about configuring and operating FortiProxy-VM after a successful deployment is available in the Fortinet Document Library.

To configure GUI access on the port1 interface:

1. In your hypervisor manager, start the FortiProxy-VM and access the console window. You might need to press *Enter* to see the login prompt.
2. At the login prompt, enter the username `admin` then press `Enter`.
3. Enter an administrator password, and then confirm the password.
4. Configure the port1 IP address and netmask:

```
config system interface
    edit port1
        set mode static
        set ip <IP address> <netmask>
        append allowaccess https
    next
end
```

5. Configure the default gateway:

```
config router static
    edit 1
        set device port1
        set gateway <ip_address>
    next
end
```

6. Optionally, configure the DNS servers:

```
config system dns
    set primary <Primary DNS server>
    set secondary <Secondary DNS server>
end
```

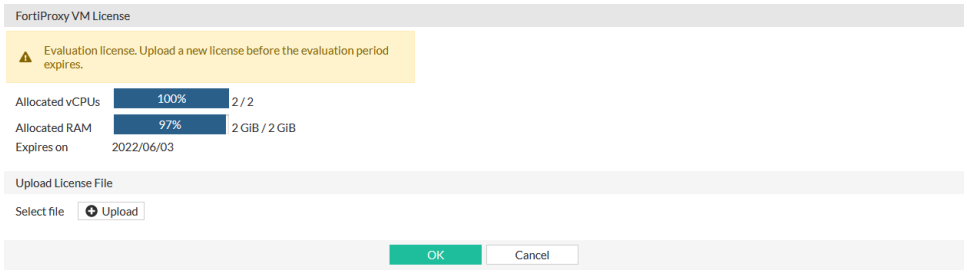
The default DNS servers are 208.91.112.53 and 208.91.112.52.

To connect to the FortiProxy-VM GUI:

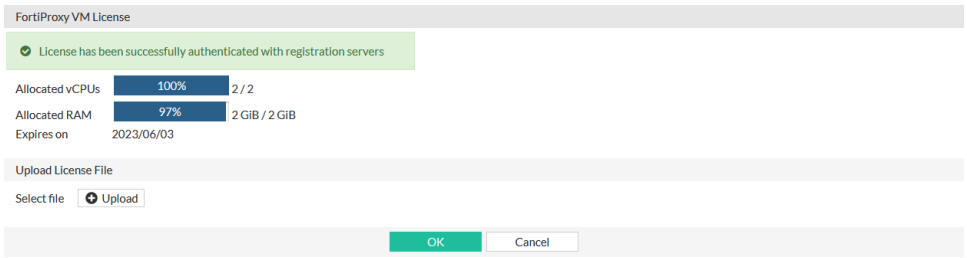
1. Launch a web browser, and enter the IP address you configured for the port1 management interface. For example: `https://192.168.0.1`.
2. At the login page, enter the username `admin` and the password that you configured.

To upload the license file:

1. Go to *System* > *FortiGuard* and click *FortiProxy-VM License*.



2. Click *Upload* and find the license file (.lic) on your computer.
3. Click *OK* to upload the license.
4. Log in to the FortiProxy-VM.
5. Confirm that the license has been successfully uploaded and validated by FortiGuard Distribution Network (FDN):
   a. Go to *Dashboard* > *Status*. The VM registration status appears as valid in the *Virtual Machine* and *Licenses* widgets
   b. Go to *System* > *FortiGuard* and click *FortiProxy-VM License*. A message reports that the license was successfully authenticated.



   c. If logging is enabled, the log message "`License status changed to VALID`" is recorded in the event log.
   d. If the update failed:
      i. Check the following settings on the FortiProxy-VM:
         - Time and time zone
         - DNS settings
         - Network interface statuses and IP addresses
         - Static routes
      ii. On the management computer, verify that FortiGuard domain names are resolving:

```
C:\>nslookup update.fortiguard.net
Server:  google-public-dns-a.google.com
Address:  8.8.8.8
```

```
Name:    fds1.fortinet.com
Addresses:  2620:101:9005:1100::205
          192.168.100.205
          192.168.100.220
Aliases:  update.fortiguard.net
```

iii. On the FortiProxy, verify that communication with the internet and FortiGuard is possible:

```
# execute ping update.fortiguard.net
PING fds1.fortinet.com (173.243.138.67): 56 data bytes
64 bytes from 173.243.138.67: icmp_seq=0 ttl=58 time=8.1 ms
64 bytes from 173.243.138.67: icmp_seq=1 ttl=58 time=3.2 ms
64 bytes from 173.243.138.67: icmp_seq=2 ttl=58 time=3.0 ms
64 bytes from 173.243.138.67: icmp_seq=3 ttl=58 time=3.8 ms
64 bytes from 173.243.138.67: icmp_seq=4 ttl=58 time=2.6 ms

--- fds1.fortinet.com ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 2.6/4.1/8.1 ms
```

```
# execute traceroute update.fortiguard.net
traceroute to update.fortiguard.net (173.243.138.67), 32 hops max, 3 probe packets per
hop, 84 byte packets
 1  192.168.0.7  10.584 ms  2.927 ms  5.073 ms
 2  10.29.206.1  5.982 ms  8.006 ms  4.199 ms
 3  154.11.11.113  3.584 ms  7.947 ms  8.679 ms
 4  154.11.2.86  2.428 ms  2.337 ms  2.645 ms
 5  * 66.163.69.46 <rd3bb-tge0-11-0-0.vc.shawcable.net>  1.586 ms  1.915 ms
 6  * 64.141.25.113 <h64-141-25-113.bigpipeinc.com>  3.491 ms  2.571 ms
 7  64.141.25.114 <h64-141-25-114.bigpipeinc.com>  1.563 ms  2.385 ms  1.966 ms
 8  96.45.47.39  2.475 ms  2.106 ms  2.105 ms
 9  173.243.138.252  2.452 ms  2.305 ms  1.877 ms
10  173.243.138.67 <update.fortiguard.net>  2.220 ms  1.620 ms  1.990 ms
```

iv. Wait for the next automatic license query (about 30 minutes), or reboot the FortiProxy-VM: `execute reboot`.

If FortiProxy is unable to validate the license after four hours a warning message it displayed in the local console.

Optimizing FortiProxy-VM performance using SR-IOV

FortiProxy-VMs installed on KVM platforms support Single Root I/O virtualization (SR-IOV) to provide FortiProxy-VMs with direct access to physical network cards. Enabling SR-IOV means that one PCIe network card or CPU can function for a FortiProxy-VM as multiple separate physical devices. SR-IOV reduces latency and improves CPU efficiency by allowing network traffic to pass directly between a FortiProxy-VM and a network card, bypassing KVM host software and without using virtual switching.

FortiProxy-VMs benefit from SR-IOV because SR-IOV optimizes network performance and reduces latency and CPU usage. FortiProxy-VMs do not use KVM features that are incompatible with SR-IOV, so you can enable SR-IOV without negatively affecting your FortiProxy-VM. SR-IOV implements an I/O memory management unit (IOMMU) to differentiate between different traffic streams and apply memory and interrupt translations between the physical functions (PF) and virtual functions (VF).

Setting up SR-IOV on KVM involves creating a PF for each physical network card in the hardware platform. Then, you create VFs that allow FortiProxy-VMs to communicate through the PF to the physical network card. VFs are actual PCIe hardware resources and only a limited number of VFs are available for each PF.

SR-IOV hardware compatibility

SR-IOV requires that the hardware and operating system on which your KVM host is running has BIOS, physical NIC, and network driver support for SR-IOV.

To enable SR-IOV, your KVM platform must run on hardware that is compatible with SR-IOV and with FortiProxy-VMs. FortiProxy-VMs require network cards that are compatible with the supported drivers. As well, the host hardware CPUs must support second level address translation (SLAT).

For optimal SR-IOV support, install the most up-to-date network drivers. Fortinet recommends i40e/Iavf drivers because they provide four TxRx queues for each VF and ixgbevf only provides two TxRx queues.

Enabling SR-IOV support for Intel systems

Use the following steps to enable SR-IOV support for KVM host systems that use Intel CPUs. These steps involve enabling and verifying Intel VT-d specifications in the BIOS and Linux kernel. You can skip these steps if you have already enabled VT-d.

On an Intel host PC, Intel VT-d BIOS settings provide hardware support for directly assigning a physical device to a VM.

To enable SR-IOV support for Intel systems:

1. View the BIOS settings of the host machine and enable VT-d settings if they are not already enabled.
   You may have to review the manufacturer's documentation for details.
2. Activate Intel VT-d in the Linsux kernel by adding the `intel_iommu=on` parameter to the kernel line in the `/boot/grub/grub.conf` file. For example:
```
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Server (2.6.32-330.x86_645)
   root (hd0,0)
   kernel /vmlinuz-2.6.32-330.x86_64 ro root=/dev/VolGroup00/LogVol00 rhgb quiet intel_iommu=on
   initrd /initrd-2.6.32-330.x86_64.img
```
3. Restart the system.

Enabling SR-IOV support for AMD systems

Use the following steps to enable SR-IOV support for KVM host systems that use AMD CPUs. These steps involve enabling the AMD IOMMU specifications in the BIOS and Linux kernel. You can skip these steps if you have already enabled AMD IOMMU.

On an AMD host PC, IOMMU BIOS settings provide hardware support for directly assigning a physical device to a VM.

To enable SR-IOV support for AMD systems:

1. View the BIOS settings of the host machine and enable IOMMU settings if they are not already enabled.
   You may have to review the manufacturer's documentation for details.
2. Append `amd_iommu=on` to the kernel command line in `/boot/grub/grub.conf` so that AMD IOMMU specifications are enabled when the system starts up.
3. Restart the system.

Verifying that Linux and KVM can find SR-IOV-enabled PCI devices

You can use the `lspci` command to view the list of PCI devices and verify that your SR-IOV supporting network cards are on the list. The following output example shows some example entries for the Intel 82576 network card:

```
# lspci
03:00.0 Ethernet controller: Intel Corporation 82576 Gigabit Network Connection (rev 01)
03:00.1 Ethernet controller: Intel Corporation 82576 Gigabit Network Connection (rev 01)
```

Optionally modifying the SR-IOV kernel modules

If the device is supported, the kernel should automatically load the driver kernel module. You can enable optional parameters using the `modprobe` command. For example, the Intel 82576 network interface card uses the `igb` driver kernel module.

```
# modprobe igb [<option>=<VAL1>,<VAL2>,
   # lsmod |grep
   igb 87592
   dca 6708 1 igb
```

Attaching an SR-IOV network device to a FortiProxy-VM

You can enable SR-IOV for a FortiProxy-VM by creating a Virtual Function (VF) and attaching the VF to your FortiProxy-VM.

Activating and verifying an SR-IOV VF

The `max_vfs` parameter of the `igb` module allocates the maximum number of VFs. The `max_vfs` parameter causes the driver to spawn multiple VFs.

Before activating the maximum number of VFs enter the following command to remove the `igb` module:

```
# modprobe -r igb
```

Restart the `igb` module with `max_vfs` set to the maximum supported by your device. For example, the valid range for the Intel 82576 network interface card is 0 to 7. To activate the maximum number of VFs supported by this device enter:

```
# modprobe igb max_vfs=7
```

Make the VFs persistent by adding `options igb max_vfs=7` to any file in `/etc/modprobe.d`. For example:

```
# echo "options igb max_vfs=7" >>/etc/modprobe.d/igb.conf
```

Verify the new VFs. For example, you could use the following `lspci` command to list the newly added VFs attached to the Intel 82576 network device. Alternatively, you can use `grep` to search for `Virtual Function`, to search for devices that support VFs.

```
# lspci | grep 82576 0b:00.0 Ethernet controller: Intel Corporation 82576 Gigabit Network Connection
      (rev 01)
0b:00.1 Ethernet controller: Intel Corporation 82576 Gigabit Network Connection(rev 01)
0b:10.0 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.1 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.2 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.3 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.4 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.5 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.6 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:10.7 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:11.0 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:11.1 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:11.2 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
```

```
0b:11.3 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:11.4 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
0b:11.5 Ethernet controller: Intel Corporation 82576 Virtual Function (rev 01)
```

Use the `-n` parameter of the `lspci` command to find the identifier for the PCI device. The PFs correspond to `0b:00.0` and `0b:00.1`. All VFs have `Virtual Function` in the description.

Verify that the devices exist with virsh

The `libvirt` service must recognize a PCI device before you can add it to a VM. `libvirt` uses a similar notation to the `lspci` output.

Use the `virsh nodedev-list` command and the `grep` command to filter the Intel 82576 network device from the list of available host devices. In the example, `0b` is the filter for the Intel 82576 network devices. This may vary for your system and may result in additional devices.

```
# virsh nodedev-list | grep 0b
pci_0000_0b_00_0
pci_0000_0b_00_1
pci_0000_0b_10_0
pci_0000_0b_10_1
pci_0000_0b_10_2
pci_0000_0b_10_3
pci_0000_0b_10_4
pci_0000_0b_10_5
pci_0000_0b_10_6
pci_0000_0b_11_7
pci_0000_0b_11_1
pci_0000_0b_11_2
pci_0000_0b_11_3
pci_0000_0b_11_4
pci_0000_0b_11_5
```

The serial numbers for the VFs and PFs should be in the list.

Get device details with virsh

The `pci_0000_0b_00_0` is one of the PFs and `pci_0000_0b_10_0` is the first corresponding VF for that PF. Use `virsh nodedev-dumpxml` to get device details for both devices.

Example device details for the `pci_0000_0b_00_0` PF device:

```
 # virsh nodedev-dumpxml pci_0000_0b_00_0
<device>
   <name>pci_0000_0b_00_0</name>
   <parent>pci_0000_00_01_0</parent>
   <driver>
      <name>igb</name>
   </driver>
   <capability type='pci'>
      <domain>0</domain>
```

```
      <bus>11</bus>
      <slot>0</slot>
      <function>0</function>
      <product id='0x10c9'>82576 Gigabit Network Connection</product>
      <vendor id='0x8086'>Intel Corporation</vendor>
   </capability>
</device>
```

Example device details for the `pci_0000_0b_10_0` PF device:

```
# virsh nodedev-dumpxml pci_0000_0b_10_0
<device>
   <name>pci_0000_0b_10_0</name>
   <parent>pci_0000_00_01_0</parent>
   <driver>
      <name>igbvf</name>
   </driver>
   <capability type='pci'>
      <domain>0</domain>
      <bus>11</bus>
      <slot>16</slot>
      <function>0</function>
      <product id='0x10ca'>82576 Virtual Function</product>
      <vendor id='0x8086'>Intel Corporation</vendor>
   </capability>
</device>
```

You must use this information to specify the bus, slot, and function parameters when you add the VF to a FortiProxy-VM. A convienent way to do this is to create a temporary xml file and copy the following text into that file.

```
<interface type='hostdev' managed='yes'>
     <source>
       <address type='pci' domain='0' bus='11' slot='16' function='0'/>
     </source>
   </interface>
```

You can also include additional information about the VF such as a MAC address, VLAN tag, and so on. If you specify a MAC address, the VF will always have this MAC address. If you do not specify a MAC address, the system generates a new one each time the FortiProxy-VM restarts.

Add the VF to a FortiProxy-VM

Enter the following command to add the VF to a FortiProxy-VM. This configuration attaches the new VF device immediately and saves it for subsequent FortiProxy-VM restarts.

```
virsh attach-device MyFGTVM <temp-xml-file> --config
```

Where *MyFGTVM* is the name of the FortiProxy-VM for which to enable SR-IOV, and `<temp-xml-file>` is the temporary XML file containing the VF configuration.

After this configuration, when you start up the FortiProxy-VM it detects the SR-IOV VF as a new network interface.

www.fortinet.com