

# Deploy the ExtraHop Discover Appliance on a Linux KVM

Published: 2020-02-23

The following procedure guides you through the deployment process of the ExtraHop Discover EDA 1000v or EDA 2000v virtual appliance on a Linux kernel-based virtual machine (KVM). You should be familiar with basic KVM administration before proceeding.

If you have not already done so, download the ExtraHop Discover virtual appliance file for KVM from the [ExtraHop Customer Portal](#).

**Important:** If you want to deploy more than one ExtraHop virtual appliance, create the new instance with the original deployment package or clone an existing instance that has never been started.

## System requirements

Your environment must meet the following requirements to deploy a Discover appliance:

EDA 1000v	EDA 2000v
A KVM hypervisor environment capable of hosting a VM that includes:	A KVM hypervisor environment capable of hosting a VM that includes:
<ul style="list-style-type: none"> <li>4 GB RAM</li> <li>Two vCPUs</li> <li>4 GB boot disk (virtio-scsi interface recommended)</li> <li>40 GB datastore disk</li> </ul>	<ul style="list-style-type: none"> <li>6 GB RAM</li> <li>Six vCPUs</li> <li>4 GB boot disk (virtio-scsi interface recommended)</li> <li>250 GB datastore disk</li> </ul>
(Optional) Open vSwitch virtual switch software	(Optional) Open vSwitch virtual switch software
An ExtraHop virtual appliance license key	An ExtraHop virtual appliance license key

## Package contents

The installation package for KVM systems is a tar.gz file that contains the following items:

Description	EDA 1000v file name	EDA 2000v file name
Domain XML configuration file	eda-1000v.xml	eda-2000v.xml
Boot disk	extrahop-boot.qcow2	extrahop-boot.qcow2
Datastore disk	extrahop-data.qcow2	extrahop-data.qcow2

## Deploy the Discover virtual appliance

To deploy the Discover virtual appliance, complete the following procedures:

- Determine the best virtual bridge configuration for your network
- Create a virtual capture bridge that contains the traffic you want to monitor
- Edit the domain XML configuration file
- Configure a mirror session on the virtual bridge

## Determine the best bridge configuration

Gather information about your network to determine the best virtual bridge configuration.

1. Identify the source of your wire data and the type of data you want to capture.
  - For SPAN, RSPAN, or port mirroring, create the virtual capture bridge with Open vSwitch.
  - For ERSPAN or rpcapd, choose either Open vSwitch or the built-in Linux bridge to create the virtual capture bridge.
2. Determine if you want to capture traffic from an external network source. If yes, configure a physical interface on the virtual capture bridge.
3. Identify the bridge you want to access the management interface through.
  - We recommend that you configure separate bridges for the capture bridge and the management bridge.
  - • The management bridge must be accessible to the ExtraHop virtual appliance and to all users who must access the management interface.
  - If you need to access the management interface from an external computer, configure a physical interface on the virtual capture bridge.

## Create the virtual capture bridge

Before you enable packet capture by an ExtraHop virtual appliance, you must create a virtual bridge that is set to promiscuous mode. If you want to capture traffic from an external network, you must add a physical interface to the bridge, and that interface must be also be set to promiscuous mode.

The following procedure describes how to create a virtual bridge with Open vSwitch. For information on how to create a virtual bridge with the built-in Linux bridge, refer to the documentation for your KVM system.

1. Log into the KVM system.
2. Create a virtual bridge by running the following command:

```
sudo ovs-vsctl add-br <bridge_name>
```

Where **<bridge\_name>** is the name of your virtual bridge.

3. Put the virtual bridge in promiscuous mode by running the following command:

```
sudo ifconfig <bridge_name> promisc
```

Where **<bridge\_name>** is the name of your virtual bridge.

4. If you want to access traffic on an external network, add a physical interface to the bridge by running the following command:

```
sudo ovs-vsctl add-port <bridge_name> <port_name>
```

Where **<bridge\_name>** is the name of your virtual bridge and **<port\_name>** is the name of the port that you want to add to the bridge.

5. If you added a physical interface to the bridge, put that interface in promiscuous mode by running the following command:

```
sudo ifconfig <port_name> promisc
```

Where **<port\_name>** is the name of the port.



**Note:** If you want the interface changes to persist after a reboot, add the ifconfig commands to your `/etc/network/interfaces` file.

## Edit the domain XML configuration file

After you create your virtual bridge, edit the configuration file, and create the ExtraHop virtual appliance.

1. Extract the tar.gz file that contains the installation package.
2. Copy the two disks `extrahop-boot.qcow2` and `extrahop-data.qcow2` to your KVM system. Make a note of the location where you store these files
3. Open the domain XML configuration file. Find and edit the following values:

- a) Change the VM name (ExtraHop-EDA-1000v or ExtraHop-EDA-2000v) to the name you want to use for your ExtraHop virtual appliance.

```
<name>ExtraHop-EDA-1000v</name>
```

- b) Change the source file path (`[PATH_TO_STORAGE]`) to the location where you stored the virtual disk files in step 1.

```
<source file='[PATH_TO_STORAGE]/extrahop-boot.qcow2' />
<source file='[PATH_TO_STORAGE]/extrahop-data.qcow2' />
```

- c) Change the source bridge for your capture network (mirrorbr0) to match the name of your capture bridge.

```
<interface type='bridge'>
  <source bridge='mirrorbr0' />
  <virtualport type='openvswitch'>
  </virtualport>
  <model type='virtio' />
  <alias name='net1' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x06'
    function='0x0' />
</interface>
```



**Note:** If you are configuring the built-in Linux bridge, remove the `virtualport type` setting.

- d) Change the source bridge for the management network (ovsbr0) to match the name of your management bridge.

```
<interface type='bridge'>
  <source bridge='ovsbr0' />
  <virtualport type='openvswitch'>
  </virtualport>
  <model type='virtio' />
  <alias name='net0' />
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03'
    function='0x0' />
</interface>
```



**Note:** If you are configuring the built-in Linux bridge, remove the `virtualport type` setting.

- e) For the EDA 2000v only: You can configure two additional bridges on the EDA 2000v. To configure another management bridge, select the configuration pattern for the ovsbr0 source bridge. To configure another capture bridge, select the configuration pattern for the mirrorbr0 source bridge.

4. Save the XML file.
5. Log in to the KVM console.

6. Create the new ExtraHop virtual appliance with your revised domain XML configuration file by running the following command:

```
virsh define <domain_XML_file>
```

Where **<domain\_XML\_file>** is the name of your domain XML configuration file (`eda-1000v.xml` or `eda-2000v.xml`)

7. Run the following command to start the VM:

```
virsh start <vm_name>
```

Where **<vm\_name>** is the name of your VM.

## Configure a mirror session on the capture bridge

This procedure explains how to configure a mirror session on an Open vSwitch virtual bridge.

1. Log in to the KVM console.
2. Export the configuration file for your new ExtraHop virtual appliance by running the following command:

```
sudo virsh dumpxml <vm_name>
```

3. In the XML output, find the name of your capture bridge. Locate the line that designates the target dev for this bridge (`<target dev = 'virtual_port_name'>`). Make a note of the virtual port name assigned to the target dev.
4. Add the virtual port to the bridge by running the following command:

```
sudo ovs-vsctl add-port <bridge_name> <virtual_port_name>
```

Where **<bridge\_name>** is the name of your capture bridge and **<virtual\_port\_name>** is the name of virtual port from the target dev setting that you noted in step 3.

5. Place this virtual port in promiscuous mode by running the following command:

```
sudo ifconfig <virtual_port_name> promisc
```

6. To monitor traffic from an external network, perform the following procedure to configure a mirror on the bridge.
  - a) Create the port mirror on the capture bridge by running the following command:

```
sudo ovs-vsctl -- --id=@m create mirror name=<your_mirror_name> -- add bridge <bridge_name> mirrors @m
```

Where **<your\_mirror\_name>** is your desired name for the mirror and **<bridge\_name>** is the name of your capture bridge.

- b) Add a physical interface to the mirror by running the following command:

```
sudo ovs-vsctl -- --id=@<mirror_port_name> get port <mirror_port_name> -- set mirror extrahop_mirror select_src_port=@<mirror_port_name> select_dst_port=@<mirror_port_name>
```

Where **<mirror\_port\_name>** is the name of the port you want to mirror.



**Note:** This example adds the port as both a source port (to capture outgoing traffic) and as a destination port (to capture incoming traffic). If you want to capture traffic in only one direction on the port, add the port as a source port (`select_src_port`) or a destination port (`select_dst_port`) only.



**Tip:** If you want to monitor only internal traffic, replace `<mirror_port_name>` with the name of the capture bridge you want to monitor.

- c) Add the virtual port name (from step 3) as the output port for the mirror by running the following command:

```
sudo ovs-vsctl -- --id=@<virtual_port_name> get port
<virtual_port_name> -- set mirror <your_mirror_name> output-
port=@<virtual_port_name>
```

## Start the VM

After you have created your new ExtraHop virtual appliance, you can log in to the management interface through a web browser to apply your license key, see network traffic, and customize your ExtraHop virtual appliance.

1. Start the VM by running the following command:  
`virsh start <vm_name>`  
 Where `<vm_name>` is the name of your ExtraHop virtual appliance.
2. Log in to the KVM console and view the IP address for your new ExtraHop virtual appliance by running the following command:

```
sudo virsh console <vm_name>
```

## Configure a static IP address

By default, ExtraHop appliances ship with DHCP enabled. If your network does not support DHCP, you must configure a static address manually.

1. Log into the KVM host.
2. Run the following command to connect to the ExtraHop appliance through the virtual serial console:

```
virsh console <vm_name>
```

Where `<vm_name>` is the name of your virtual machine.

3. Press ENTER twice to get to the appliance login prompt.

```
ExtraHop Discover Appliance Version 6.2.6.3385
IP: 192.0.2.81
exampleium login:
```

4. At the login prompt, type `shell`, and then press ENTER.
5. At the password prompt, type `default`, and then press ENTER.
6. To configure the static IP address, run the following commands:
  - a) Enable privileged commands:

```
enable
```

- b) At the password prompt, type `default`, and then press ENTER.
  - c) Enter configuration mode:

```
configure
```

- d) Enter the interface configuration mode:

```
interface
```

- e) Run the `ip` command and specify the IP address and DNS settings in the following format: `ip ipaddr <ip_address> <netmask> <gateway> <dns_server>`  
For example:

```
ip ipaddr 10.10.2.14 255.255.0.0 10.10.1.253 10.10.1.254
```

- f) Leave the interface configuration section:

```
exit
```

- g) Save the running config file:

```
running_config save
```

- h) Type `y` and then press ENTER.

## Configure the Discover appliance

After you configure an IP address for the Discover appliance, open a web browser and navigate to the ExtraHop Web UI through the configured IP address. Accept the license agreement and then log in. The default login name is **setup** and the password is **default**. Enter the product key to license the appliance.

After the appliance is licensed, and you have verified that traffic is detected, complete the recommended procedures in the [post-deployment checklist](#).