

# Deploy an ExtraHop sensor on Linux KVM

Published: 2024-04-04

The following procedure guides you through the deployment process of the ExtraHop EDA 1100v virtual sensor 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 virtual sensor file for KVM from the [ExtraHop Customer Portal](#).

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

## Virtual machine requirements

Your KVM hypervisor must be able to support the following specifications for the virtual sensor.

Sensor	vCPU	RAM	Disk
Reveal(x) EDA 1100v	2	4 GB	<ul style="list-style-type: none"> <li>4 GB boot disk (virtio-scsi interface recommended)</li> <li>40 GB datastore disk</li> <li>(Optional) 250 GB or smaller disk for packet captures (thick-provisioned)</li> </ul>

The hypervisor CPU should provide Streaming SIMD Extensions 4.2 (SSE4.2) and POPCNT instruction support.

**Note:** If you want to enable packet captures, configure an additional storage disk during deployment. Refer to your vendor documentation to add a disk.

## Package contents

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

Description	Reveal(x) 1100v
Domain XML configuration file	eda-1100v.xml
Domain XML checksum file	eda-1100v.xml.md5
Boot disk	extrahop-boot.qcow2
Boot disk checksum file	extrahop-boot.qcow2.md5
Datastore disk	extrahop-data.qcow2
Datastore disk checksum file	extrahop-data.qcow2.md5

## Deploy the virtual sensor

To deploy the virtual sensor, 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 virtual sensor 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 sensor, 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 in to 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.

- 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 sensor.

- Extract the tar.gz file that contains the installation package.
- 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
- Open the domain XML configuration file. Find and edit the following values:
  - Change the VM name (ExtraHop-EDA-1100v) to the name you want to set for your ExtraHop virtual sensor.

```
<name>ExtraHop-EDA-1100v</name>
```

- 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' />
```

- 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.

- 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.

4. Save the XML file.
5. Log in to the KVM console.
6. Create the new ExtraHop virtual sensor 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-1100v.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.

### Before you begin

 **Important:** To ensure the best performance for initial device synchronization, connect all sensors to the console and then configure network traffic forwarding to the sensors.

1. Log in to the KVM console.
2. Export the configuration file for your new ExtraHop virtual sensor 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 <your_mirror_name> 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 and *<your\_mirror\_name>* is the name you specified in step 6a.



**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 ExtraHop virtual sensor, you can log in to the management interface through a web browser to apply your license key, see network traffic, and customize your sensor configurations.

1. Start the VM by running the following command:

```
virsh start <vm_name>
```

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

2. Log in to the KVM console and view the IP address for your new ExtraHop sensor by running the following command:

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

## (Optional) Configure a static IP address

By default, the ExtraHop system is configured with DHCP enabled. If your network does not support DHCP, you must configure a static address manually.

1. Log in to the KVM host.
2. Run the following command to connect to the ExtraHop system 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 system login prompt.

```
ExtraHop Discover Appliance Version 7.8.2.2116
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 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 interface configuration mode:

```
exit
```

- g) Save the running configuration file:

```
running_config save
```

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

## Configure the sensor

### Before you begin

Before you can configure the sensor, you must have already configured a management IP address.

1. Log in to the Administration settings on the ExtraHop system through `https://<extrahop-hostname-or-IP-address>/admin`.

The default login name is `setup` and the password is the VM instance ID. The default login name is `setup` and the password is `default`.

2. Accept the license agreement and then log in.
3. Follow the prompts to enter the product key, change the default setup and shell user account passwords, connect to ExtraHop Cloud Services, and connect to an ExtraHop console.

### Next steps

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