Hyper-V Virtual Sensor (vSensor) Deployment Guide

Version: March 5, 2025

Table of Contents

Introduction

This guide is intended to help customers or partners deploy vSensors in Hyper-V environments and pair them to your Vectra Brain. It will cover basic background information, connectivity requirements (firewall rules that may be needed in your environment), deployment of the vSensor in Hyper-V, and pairing.

vSensors behave much in the same way that physical Sensors do. One advantage is that there is no cost to deploy a vSensor other than your own costs to provide and maintain the infrastructure they run in. vSensors also allow you to capture and analyze traffic that only exists in the virtual environment. You can even use vSensors in place of physical Sensors to capture physical network traffic.

Hyper-V vSensors can be used in both Respond UX and Quadrant UX deployments. For more detail on Respond UX vs Quadrant UX please see Vectra Analyst User Experiences (Respond vs Quadrant). One of the below guides should be the starting point for your overall Vectra deployment:

- Vectra Respond UX Deployment Guide
- Vectra Quadrant UX Deployment Guide

Either of the above guides cover basic firewall rules needed for the overall deployment and initial platform settings. Virtual Sensor (VMware, Hyper-V, KVM, AWS, Amazon, and GCP) configuration and pairing and covered in their respective guides. Physical appliance pairing is covered in the Vectra Physical Appliance Pairing Guide. Please see the Vectra Product Documentation Index on the Vectra support site for additional documentation including deployment guides for CDR for M365 / IDR for Azure AD and CDR for AWS.

About Hyper-V vSensor Images

The Brain makes a Hyper-V VHDX image (in a .zip archive) available for download and subsequent use for provisioning vSensors. Vectra appliances typically operate with updates enabled. Regular updates ensure that the appliances are running the very latest version. Deployed Sensors and vSensors also update regularly. When a Brain updates, the corresponding VHDX which it makes available for download and use on vSensors is updated as well. As such, it is recommended that you download and use the very latest available VHDX from your Brain any time you deploy a new vSensor. Once a vSensor has been deployed, it will update itself as needed, staying current with its Brain.

Hyper-V vSensor Requirements and Throughput

Hyper-V Version Supported	Windows Server 2016 w/ HW v8 or higher
Cores Required	2 (500 Mbps), 4 (1 Gbps), 8 (2 Gbps), or 16 (5 Gbps)
RAM Required	8 GB (2 and 4 core), 16 GB (8 core), 64 GB (16 core) Must be contiguous in a single NUMA node (not spanning multiple nodes)
Disk Space Required	100 GB (2 core), 150 GB (4 and 8 core), 500 GB (16 core)
Virtual Switch Type Supported	External
Interfaces Supported	Up to 2 for capture, 1 for management (can be shared with capture)
Traffic that can be captured	Physical or Virtual

- If you wish to resize the vSensor after deployment, please see: Resizing Virtual Sensors and Brains for details. Please note that you can only from a smaller configuration to a larger one. NOT from a larger configuration to a smaller one.
- To add a 2nd capture interface if you originally deployed with only 1, you must shut the vSensor down, add the 2nd capture interface, and restart it for this to work.
- In Hyper-V, if the processor is supported but the sensor health check is reporting "*[FAILED] Hypervisor CPU Supported*" and not capturing traffic, try disabling the processor compatibility in the settings.

Connectivity Requirements

The Vectra Respond UX Deployment Guide or Vectra Quadrant UX Deployment Guide detail basic connectivity requirements for initial platform deployment. It also gives guidance on firewall/proxy SSL inspection, Internet access to and from the Brain, and guidance for air-gapped environments. For full detail on all possible firewall rules that might be required in your environment, please see Firewall Requirements for Vectra Appliances. Hyper-V vSensor specific requirements are listed below:

Connectivity Requirements for Hyper-V vSensors

Source	Destination	Protocol/Port	Description
Admin Hosts	vSensors	TCP/22 (SSH)	CLI access to vSensor
Brain	vSensors	TCP/22 (SSH)	Remote management and troubleshooting
vSensors	Brain	TCP/22 (SSH) TCP/443 (HTTPS)	Pairing, metadata transfer, and ongoing communication

Please note:

- vSensors do not communicate with the Vectra Cloud.
- All communication sessions with vSensors are initiated from the vSensor to the Brain.
- Updates for vSensors are downloaded to the Vectra Brain and the vSensor retrieves them from the Brain.
- CLI access can also be done via the console in your hypervisor.

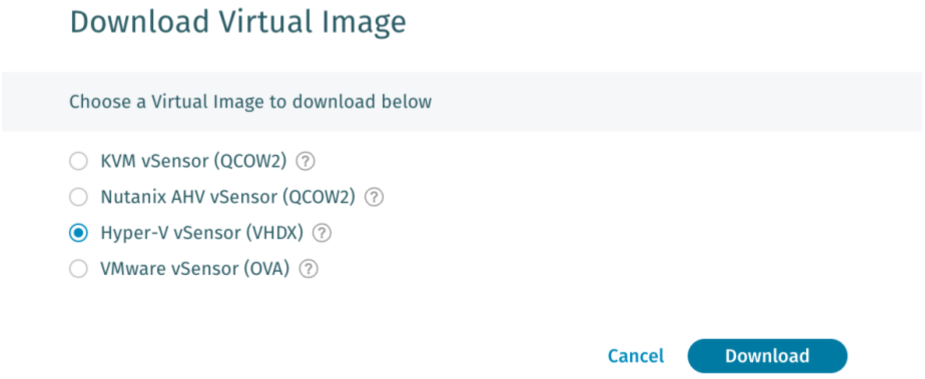
vSensor Deployment in Hyper-V

Requirements

- IP address and subnet mask for the Management interface of the vSensor.
- DNS server addresses.
- Access to PowerShell as an administrator on the Hyper-V server you plan to deploy the vSensor on.
- To configure your vSensor, you will need access to the vSensor CLI either via the console in your hypervisor or via SSH.
  - DHCP is enabled by default upon vSensor initial boot.
  - You must know the IP that was assigned via DHCP to SSH to the CLI, otherwise you will need to use the hypervisor console.

Downloading the latest vSensor Hyper-V VHDX image

The current vSensor image (VHDX) for use on new vSensors can be downloaded from the Brain by clicking the blue “Download Virtual Image” link at the top right from the *Data Sources > Network > Sensors* page in your UI and then selecting the Hyper-V vSensor (VHDX) image.



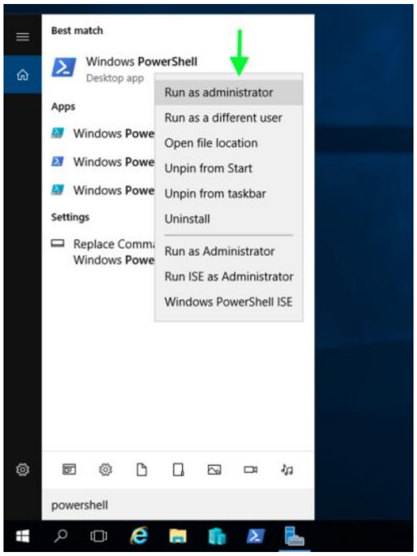
Download Virtual Image

Choose a Virtual Image to download below

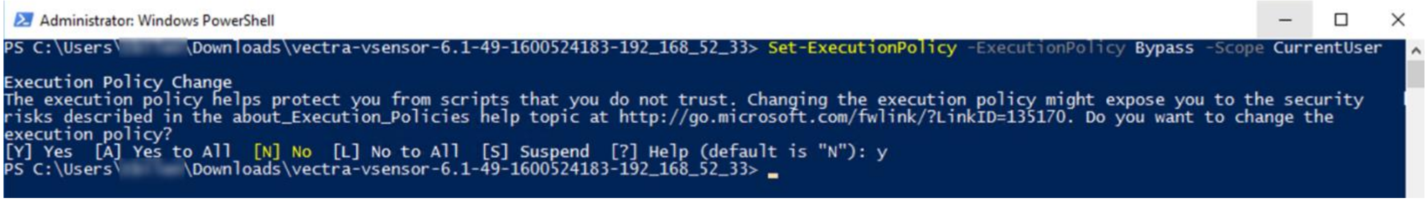
- ○ KVM vSensor (QCOW2)
- ○ Nutanix AHV vSensor (QCOW2)
- ● Hyper-V vSensor (VHDX)
- ○ VMware vSensor (OVA)

Deploying the VHDX

- Once downloaded, the VHDX image will need to be unzipped. Use the archive utility of your choice to unzip the image.
- Launch PowerShell as an administrator.



- As a ONE-TIME-SETUP STEP, run **"Set-ExecutionPolicy -ExecutionPolicy Bypass -Scope CurrentUser"** to allow running of the installer.



```
PS C:\Users\[redacted]\Downloads\vectra-vsensor-6.1-49-1600524183-192_168_52_33> Set-ExecutionPolicy -ExecutionPolicy Bypass -Scope CurrentUser

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose you to the security
risks described in the about_Execution_Policies help topic at http://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the
execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): y
PS C:\Users\[redacted]\Downloads\vectra-vsensor-6.1-49-1600524183-192_168_52_33> _
```

- Change directory to where you unzipped the VHDX image.
- You can use the PowerShell **get-help** command to see current options for the installation script
    - **"get-help '.\Vectra vSensor.ps1'"**
    - For example, you may want to use a specific name for this vSensor using the "-Name" argument.
- **Please NOTE**
    - You can specify arguments for the script and automate the deployment or do the deployment interactively and the script will prompt you for required information.
    - Some of the more common arguments for the CLI command are shown below:

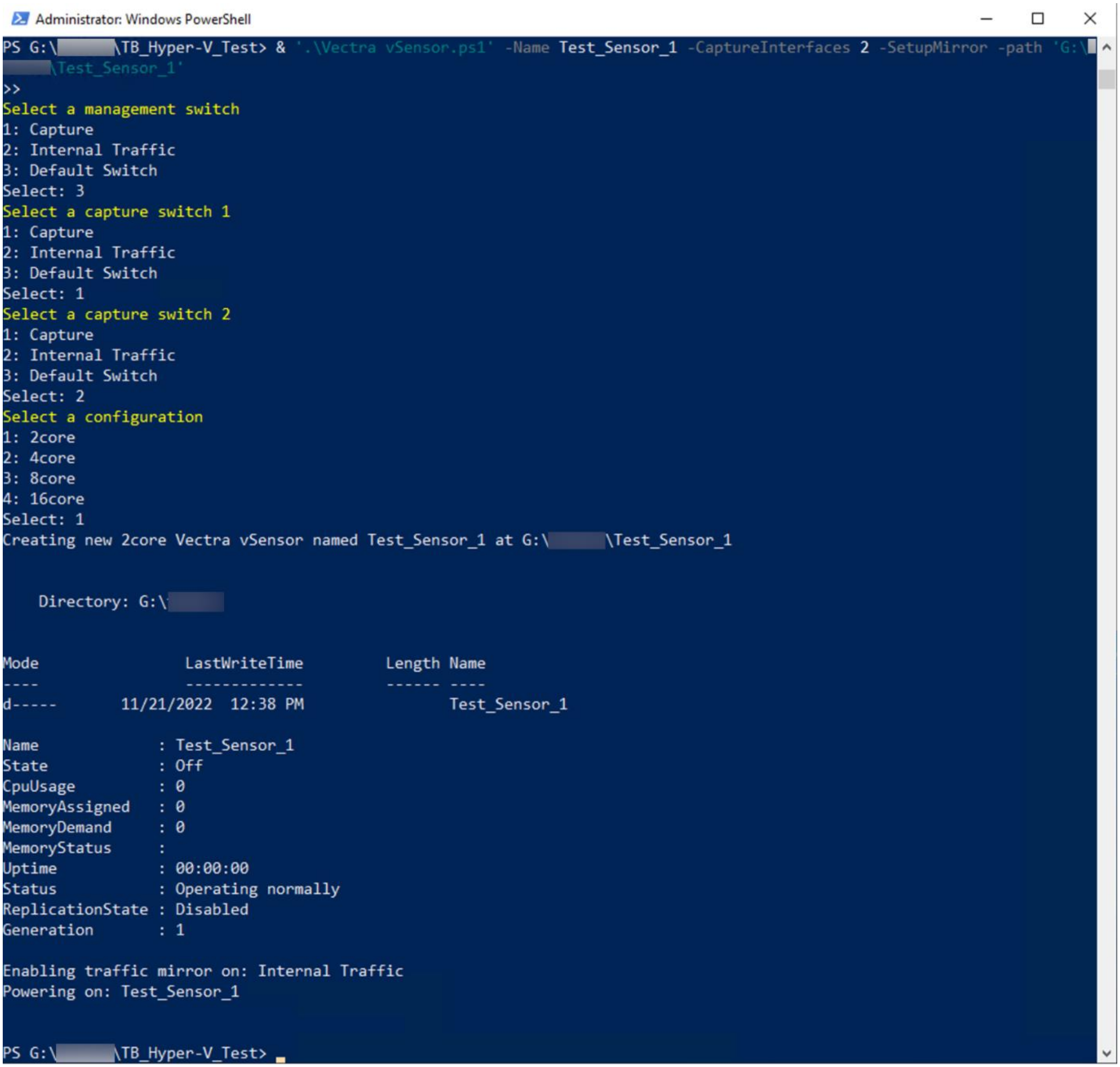
```
-Name
        The name of the new VM.

-Path
        The storage path for the VM.

-ManagementSwitch
        The network switch through which the sensor will pair with the brain.

-CaptureSwitches
        The network switches through which the sensor will ingest traffic. A comma separated list (Length
must match number of capture interfaces)

-CaptureInterfaces
        The number of capture interfaces to set up using the interfaces given in CaptureSwitches

-SetupMirror
        Setup port mirroring on the external capture vSwitch.

-Configuration
        The selected sensor configuration.

        Type: String
        Options: “2core", "4core", “8core”, “16core”
```

    - To specify the storage path for the VM you must use the **"-path"** option, or your installation will occur in the current directory.
        - If you install in the current directory, to install another vSensor you will need to unzip a new copy of the vSensor code to deploy as the .vhdx file will be in use by the 1st vSensor you deployed.
        - It is a best practice to specify a path for installation if you will be installing multiple vSensors.
    - Remote hosts are also supported, for guidance in specifying remote paths, refer to the **"get-help"** option.
    - The **"-setupmirror"** option will attempt to setup port mirroring on any capture switch. It does this by setting "Monitor Mode 2" on the capture switch as the source and should be used if you want to capture the traffic between the virtual switch and the external NIC (capturing physical traffic for assets outside of Hyper-V).
        - This will cause all traffic passing on the external network NIC of the capture switch to be mirrored to any VM whose port monitoring mode has been set to **"Destination".**
        - If you already have this set, there is no need to set it again.
        - Hyper-V switches that don't support Monitor Mode 2 (such as "Internal" Hyper-V switches) cannot be set to Monitor Mode 2 but will not error out or cause issues.
        - See Capturing physical network traffic coming into the Hyper-V server for more detail.
- Execute the PowerShell script to begin the installation.
    - **"& '.\Vectra vSensor.ps1'"** (and optionally use **"-setupmirror**" as a switch if desired)
    - You will need to specify the management switch, capture switch, and configuration.
    - The script will then create the vSensor and start it up.

This below example is interactively deploying a new vSensor. The arguments have supplied:

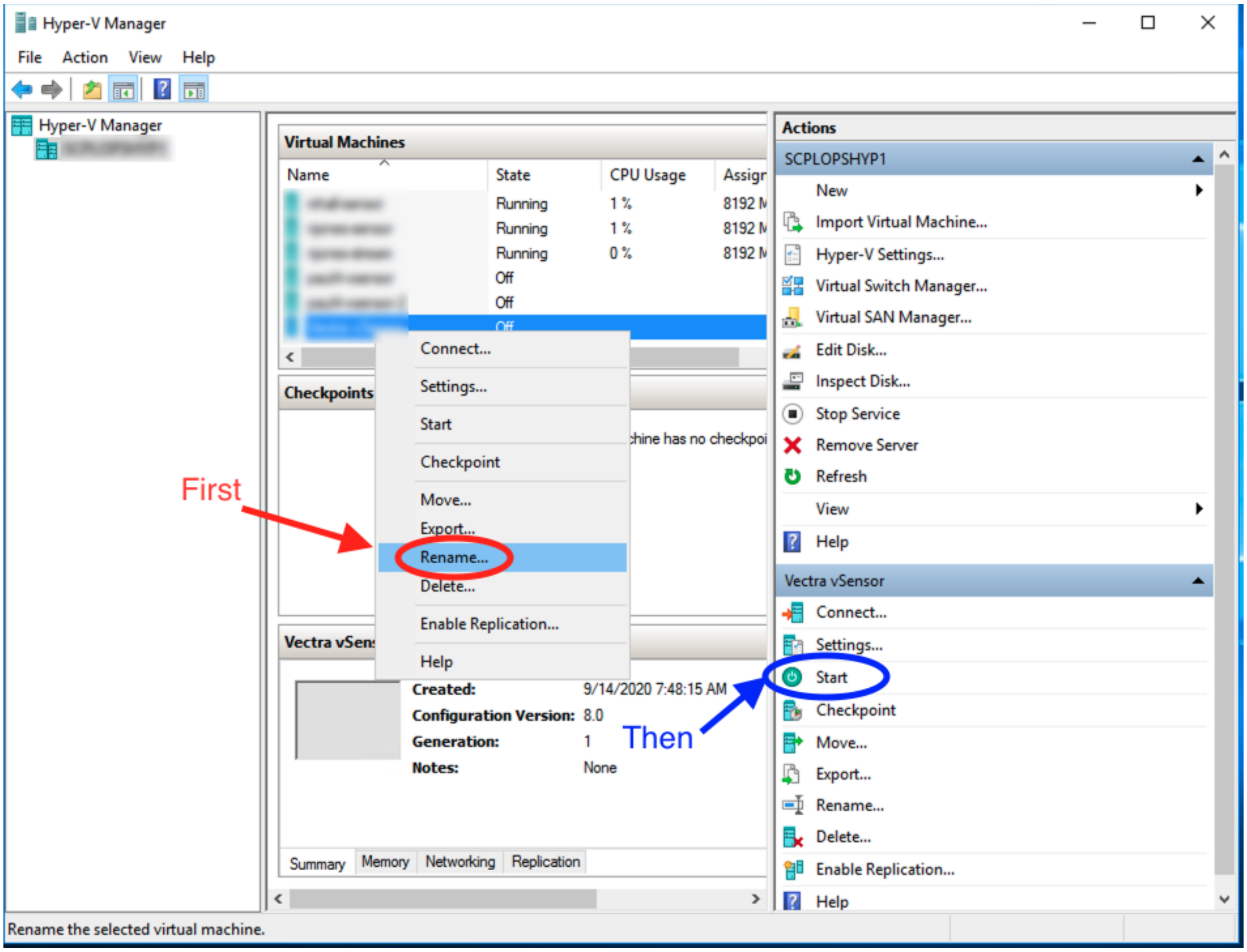
- A name of “Test_Sensor_1”
- A path of “G:\redacted\Test_Sensor_1”
- We’ve asked the script to configure 2 capture interfaces on the vSensor but will specify the Hyper-V switches we can to map to interactively.

Test_Sensor_1” directory:



```
PS G:\[redacted]\TB_Hyper-V_Test> & '.\Vectra vSensor.ps1' -Name Test_Sensor_1 -CaptureInterfaces 2 -SetupMirror -path 'G:\
[redacted]\Test_Sensor_1'
>>
Select a management switch
1: Capture
2: Internal Traffic
3: Default Switch
Select: 3
Select a capture switch 1
1: Capture
2: Internal Traffic
3: Default Switch
Select: 1
Select a capture switch 2
1: Capture
2: Internal Traffic
3: Default Switch
Select: 2
Select a configuration
1: 2core
2: 4core
3: 8core
4: 16core
Select: 1
Creating new 2core Vectra vSensor named Test_Sensor_1 at G:\[redacted]\Test_Sensor_1


    Directory: G:\[redacted]


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----       11/21/2022  12:38 PM                Test_Sensor_1

Name             : Test_Sensor_1
State            : Off
CpuUsage         : 0
MemoryAssigned   : 0
MemoryDemand     : 0
MemoryStatus     :
Uptime           : 00:00:00
Status           : Operating normally
ReplicationState : Disabled
Generation       : 1

Enabling traffic mirror on: Internal Traffic
Powering on: Test_Sensor_1


PS G:\[redacted]\TB_Hyper-V_Test> _
```

- Once the script has finished you can rename your VM in the Hyper-V manager if desired.
  - The vSensor will start automatically after the deployment script runs.
  - Renaming can be done while the vSensor is running.
  - If you shut down the vSensor, the Start button location is shown below.



Name	State	CPU Usage	Assign
[redacted]	Running	1 %	8192 M
[redacted]	Running	1 %	8192 M
[redacted]	Running	0 %	8192 M
[redacted]	Off		
[redacted]	Off		
[redacted]	Off		

chine has no checkpoi

Created: 9/14/2020 7:48:15 AM
Configuration Version: 8.0
Generation: 1
Notes: None

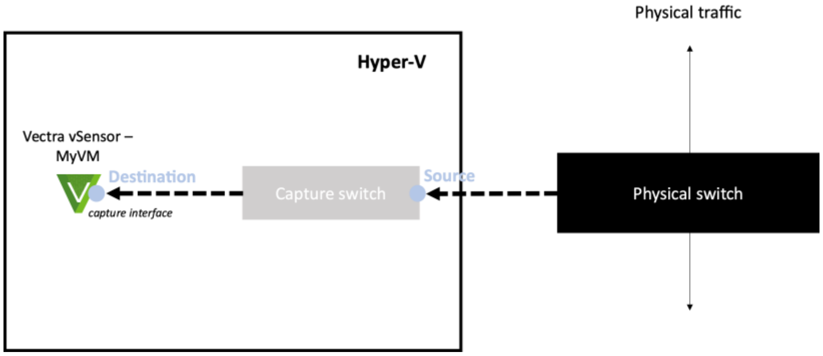
Rename the selected virtual machine.

Capture Configuration Guidance and Virtual Switch Options

Hyper-V vSensors can capture virtual guest network traffic, physical network traffic, and support VLAN tagging. This section will detail settings, include some sample syntax, and share some best practices.

- **NOTE:** Up to 2 interfaces per vSensor can be set for capture.
  - A capture switch can be the same as the management switch if your server does not have a dedicated capture switch.
- If you have physical network traffic directed at a switch that is not being used by your guests running on the server, you can either deploy two vSensors or use 2 separate capture interfaces for each traffic type.
  - You must ensure that the vSensor is adequately sized for the expected total traffic volume.
  - Some customers may consider it a best practice to separate physical and virtual network traffic capture into two vSensors regardless of technical capability to do so.
  - Please keep in mind resource requirements for each vSensor and guest VMs when deploying multiple vSensors on the same physical host.

Capturing physical network traffic coming into the Hyper-V server



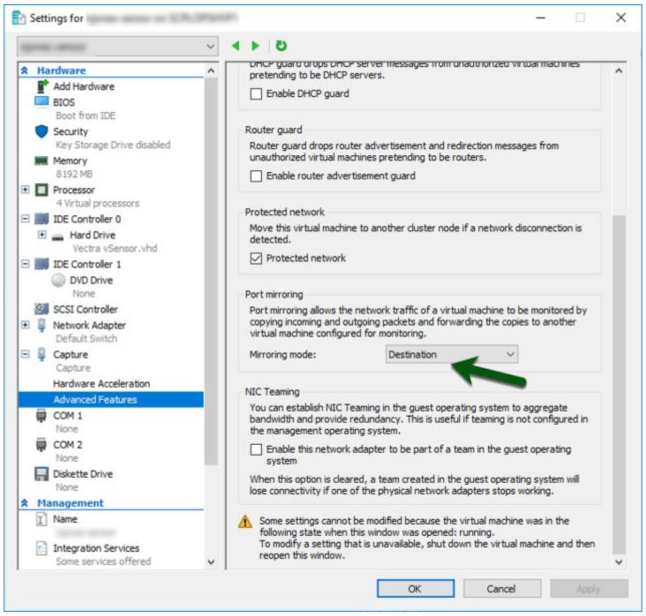
- The **“-setupmirror**” option described earlier should have been used during initial vSensor deployment if were intending to capture physical network traffic and will enable MonitorMode 2 which is essentially the same as source mode for port mirroring on the capture switch.
  - MonitorMode 2 = Source, MonitorMode 1 = Destination, MonitorMode 0 = None
  - If you did not use the -setupmirror option during the vSensor deployment you can use the following PowerShell commands to configure this on an already deployed vSensor:

```
$PortFeature = Get-VMSystemSwitchExtensionPortFeature -FeatureName "Ethernet Switch Port Security Settings"
$PortFeature.SettingData.MonitorMode = 2
Add-VMSwitchExtensionPortFeature -ExternalPort -SwitchName "$CaptureSwitch" -VMSwitchExtensionFeature $portFeature
```

    - In the above commands, $CaptureSwitch would need to be replaced with the switch name.
- You will still need to ensure that the virtual network adapter being used by the vSensor VM for the capture switch has its port mirroring mode set to **"Destination".**
  - Via PowerShell
    - Set vSensor capture interface as **“Destination”,** were MyVM is the VM name and XXXX is the MAC address of the interface.

```
Get-VMNetworkAdapter MyVM | ? MacAddress -eq 'xxxxxxxx' | Set-VMNetworkAdapter MyVM -
PortMirroring Destination
```

  - Via Hyper-V GUI
    - Note that this screenshot is an example, you will likely need to do this on the 2nd interface if using a separate management and capture interface.



Settings for [redacted]

[redacted]

Hardware
- Add Hardware
- BIOS
  Boot from IDE
- Security
  Key Storage Drive disabled
- Memory
  8192 MB
- Processor
  4 Virtual processors
- IDE Controller 0
  - Hard Drive
    Vectra vSensor.vhd
- IDE Controller 1
  - DVD Drive
    None
- SCSI Controller
- Network Adapter
  Default Switch
- Capture
  Capture
  - Hardware Acceleration
  - Advanced Features
- COM 1
  None
- COM 2
  None
- Diskette Drive
  None

Management
- Name
  [redacted]
- Integration Services
  Some services offered

DHCP guard drops DHCP server messages from unauthorized virtual machines pretending to be DHCP servers.
☐ Enable DHCP guard

Router guard
Router guard drops router advertisement and redirection messages from unauthorized virtual machines pretending to be routers.
☐ Enable router advertisement guard

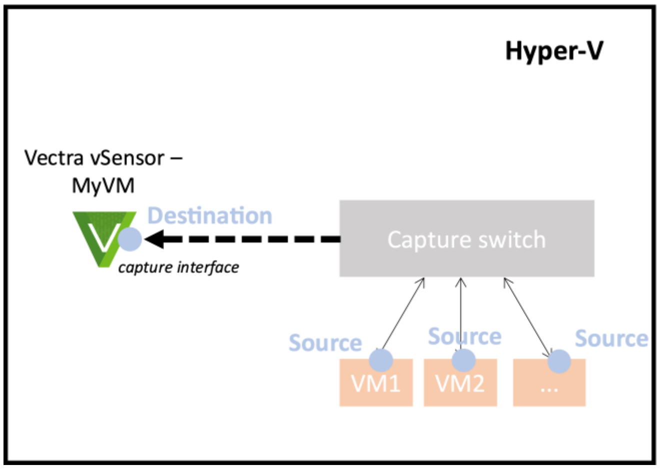
Protected network
Move this virtual machine to another cluster node if a network disconnection is detected.
☑ Protected network

Port mirroring
Port mirroring allows the network traffic of a virtual machine to be monitored by copying incoming and outgoing packets and forwarding the copies to another virtual machine configured for monitoring.
Mirroring mode: Destination

NIC Teaming
You can establish NIC Teaming in the guest operating system to aggregate bandwidth and provide redundancy. This is useful if teaming is not configured in the management operating system.
☐ Enable this network adapter to be part of a team in the guest operating system
When this option is cleared, a team created in the guest operating system will lose connectivity if one of the physical network adapters stops working.

⚠ Some settings cannot be modified because the virtual machine was in the following state when this window was opened: running.
To modify a setting that is unavailable, shut down the virtual machine and then reopen this window.

Capturing virtual network traffic from other guests



- Set the port mirroring mode on the VMs you wish to mirror as **"Source".**
- This can be done in the GUI by going to each source host or from the PowerShell by getting the relevant VM name and MAC address for the source VM ports and setting the Port Mirroring mode as "Source" instead of "Destination". Example PowerShell commands below (customized for your deployment):

```
Get-VMNetworkAdapter MyOtherVM | ? MacAddress -eq 'xxxxxxxx' | Set-VMNetworkAdapter MyOtherVM -
PortMirroring Source
```

- **Note:** Hyper-V only supports port mirroring on the individual guest VM NICs (where they need to be set up as a source).
  - It cannot simply be enabled at the switch level to capture all copies of internal inter-VM traffic and must be done at individual VM NIC level.

Capturing traffic that flows over multiple VLANs

- Whether capturing physical or inter-VM traffic, if the vSensor receives flows over multiple VLANs, then this will require the Destination host (vSensor in this case) to allow all relevant VLANs and be set for Trunking. This is done via PowerShell commands that will need to be customized for your deployment.

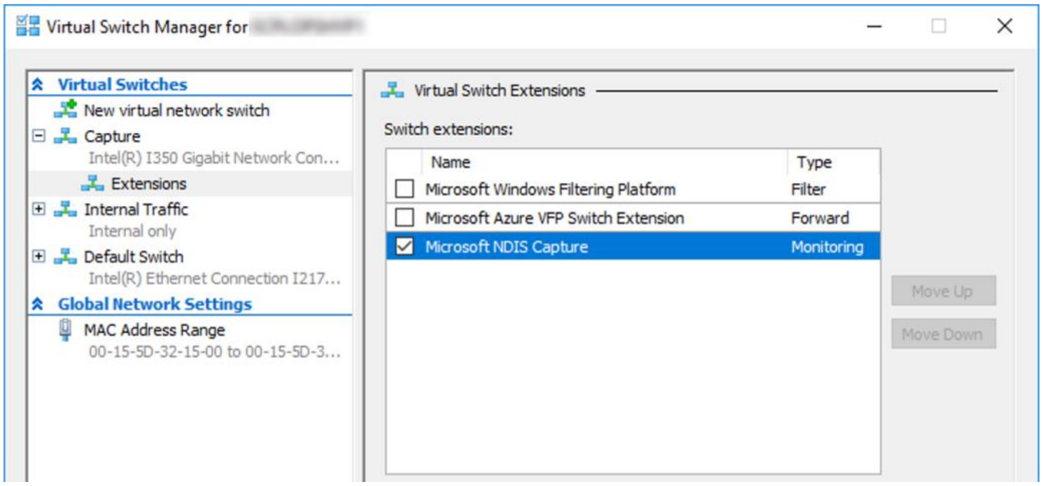
```
Set-VMNetworkAdapterVlan -VMName MyVM -VMNetworkAdapterName "mirror" -trunk -
allowedvlanidlist <VLAN-ID-Range> -nativevlanid <VLAN-ID-Range>
```

- Here are a couple of examples that show a full command.
  - The NativeVlanID parameter tells Hyper-V that if there is no VLAN specified in the packet, to treat the packet as if it was from VLAN 0 or 10 in the below example.
    - If you do not specify a specific adapter, the command will apply the settings to all adapters in the VM it is targeted to. This is not desired so please be sure to specify the adapter as per the example below (use the right name for your adapter).

```
Set-VMNetworkAdapterVlan -VMName MyVM -VMNetworkAdapterName "mirror" -Trunk -AllowedVlanIdList "100,101" -NativeVlanId 0

Set-VMNetworkAdapterVlan –VMName MyVM -VMNetworkAdapterName "mirror" –Trunk –AllowedVlanIdList 1-100 –NativeVlanId 10
```

Additional guidance for virtual switch options in Hyper-V

- Microsoft NDIS Capture must be enabled on the capture switch
- This can be set per the screenshot below in your Virtual Switch Extensions for the capture switch if it is not already enabled



Virtual Switch Manager for [redacted]

- Virtual Switches
  - New virtual network switch
  - Capture
    - Intel(R) I350 Gigabit Network Con...
    - Extensions
  - Internal Traffic
    - Internal only
  - Default Switch
    - Intel(R) Ethernet Connection I217...
- Global Network Settings
  - MAC Address Range
    - 00-15-5D-32-15-00 to 00-15-5D-3...

Virtual Switch Extensions

Switch extensions:

	Name	Type
☐	Microsoft Windows Filtering Platform	Filter
☐	Microsoft Azure VFP Switch Extension	Forward
☑	Microsoft NDIS Capture	Monitoring

- Other virtual switch hardware acceleration or advanced features can set as desired and should not impact vSensor function

Initial vSensor Configuration at CLI

- Connect to your vSensor CLI using your hypervisor console or “**`ssh vectra@<IP or Hostname>`**” if you use DHCP and already know the address or hostname.
  - If the vSensor has shown up in your UI then you should be able to see its IP address in *Data Sources > Manage > Sensors* as well.
- Once logged in to the appliance you can view command syntax for the "set interface" command:

```
set interface -h
Usage: set interface [OPTIONS] [mgt1] [dhcp|static] [IP] [SUBNET_MASK]
[GATEWAY_ADDRESS]

Sets mgt1 to either dhcp or static ip configuration

Options:
-h, --help Show this message and exit.
```

- Setting the IP address statically:
  - In v8.5 and higher of Vectra software, IPv6 is supported for the MGT1 interface. For full details, including information regarding dual stack support, please IPv6 Management Support for Vectra Appliances on the Vectra support portal. Below we will show how to enable IPv6 support (its off by default) and the syntax to use when setting an IPv4 or IPv6 address.

  - To enable/disable IPv6 support

```
# show ipv6 enabled
IPv6 is disabled

# set ipv6 enabled
Response: ok
```

```
# show ipv6 enabled
IPv6 is enabled

# set ipv6 disabled
Response: ok
```

  - Setting IPv4 and IPv6 syntax examples:

    Execute the following command to set the MGT1 interface to the desired static IP address:

    ```
    IPv4 Syntax:
    set interface mgt1 static x.x.x.x y.y.y.y z.z.z.z

    Where:
    x.x.x.x is the desired interface IP address
    y.y.y.y is the desired interface network mask
    z.z.z.z is the desired gateway

    IPv6 Syntax:
    set interface mgt1 static [IPv6 IP] [Subnet Mask] [Gateway]

    Example:
    set interface mgt1 static 2001:0db8:0:f101::25 64 2001:0db8:0:f101::1
    ```

- To change back to DHCP (default):

  ```
  set interface mgt1 dhcp
  ```

- Configure DNS for the appliance:

  Command syntax to set DNS (up to 3 nameservers are supported):

  ```
  set dns [nameserver1 <ip>] [nameserver2 <ip>] [nameserver3 <ip>]
  ```

  Example:

  ```
  set dns 10.50.10.101 10.50.10.102
  ```

  Verifying DNS Configuration:

  ```
  show dns
  ```

- Once you have set an IP and DNS, please use the "`set password`" command to change the password or you may wait and change all paired Sensor passwords en masse in the Vectra UI later at *Data Sources > Network > Sensors > Sensor Configuration > CLI Password (Sensors)* if you wish to keep them in sync.

Example:

```
ssh vectra@172.16.12.11
Password: xxxxx
Welcome to Cognito 6.5.1-1-25, up 24 weeks, 15 hours, 46 minutes (5.4.0-45-generic)

Open source licensing information used in this product is available at https://www.vectra.ai/opensource

Last login: Tue Mar 16 19:06:19 2021 from 172.16.12.1

Welcome to the Vectra Support CLI!

  Model:             DCS
  Mode:              sensor
```

```
  Update version:   6.5.1-1-25
  Colossus version: 2:6.5-148-g36b896dc41
  User:             vectra
  Local time:       2021-03-18 18:25:53.223004

  Use 'show commands' to get a list of available commands
  Use 'help' command or '<command> --help' to get help

  Welcome to the Vectra Mobile Attack Lab.  VECTRA CONFIDENTIAL

vscli > set interface mgt1 static 172.16.12.11 255.255.255.0 172.16.12.1
Interfaces updated successfully
vscli > set dns 10.50.10.101
DNS Set: success
vscli > show interface
mgt1:
    Running:
        Gateway: 172.16.12.1,
        Ip: 172.16.12.11,
        Link Speed: 10Gbps,
        Link State: up,
        Mac: 00:0c:29:89:ad:a6,
        Mode: static,
        Netmask: 255.255.255.0
vscli > show dns
Id|Server       |Description
1  10.50.10.101 Configured DNS nameserver
```

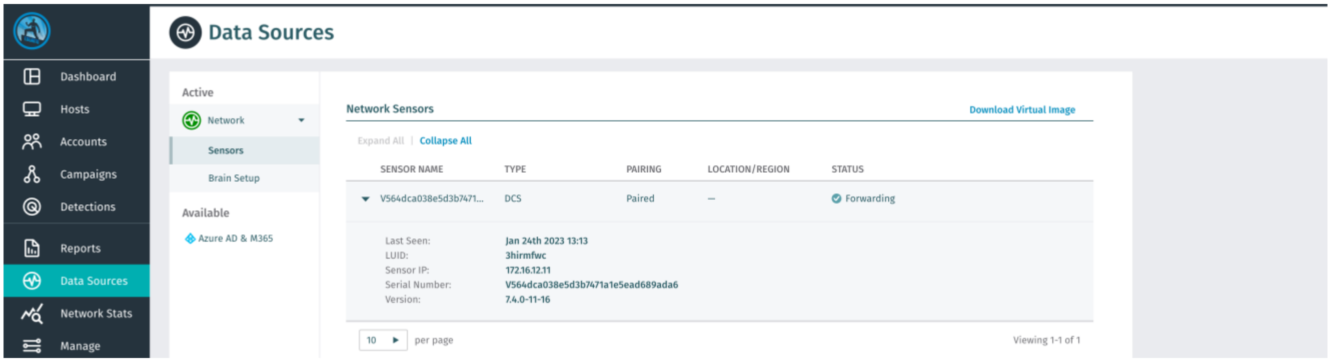
Pairing vSensors

- vSensors do not offer a web UI.
    - The GUI for vSensor management is located at *Data Sources > Network > Sensors*.
    - Some configuration of the vSensor can be done at the CLI of the vSensor using the **“vectra”** user.
- vSensor images are already associated with the Brain they were downloaded from and will appear in the Brain *Data Sources > Network > Sensors* page when booted for the first time.
- vSensors are unique to the Brain that the image was downloaded from and cannot normally be paired to other Brains in your environment.
    - If a **“Sensor Registration Token”** is used, a deployed vSensor can be paired to a Brain other than the one the image was originally downloaded from.
    - This is covered in the below Additional Pairing Guidance section below.
- As per the Vectra Respond UX Deployment Guide or Vectra Quadrant UX Deployment Guide, Sensors, including vSensors, support pairing by IP or by hostname.
    - Pairing by hostname is preferred in failover scenarios. See guide above for more details.
- Once the vSensor is powered on and the interface configuration is set, the vSensor will announce itself to the Brain.
    - This can take a couple of minutes, check firewall rules if there is an issue.
    - If the vSensor appears in the Brain *Data Sources > Network > Sensors* page, then it has made a successful HTTPS connection to the Brain.
    - If the vSensor does not appear in the Brain *Data Sources > Network > Sensors* page, check that the vSensor has IP connectivity and that TCP port 443 (HTTPS) is permitted through your firewall.
    - If the vSensor is unable to pair with the Brain, complete its initial update or forward metadata to the Brain check that TCP port 22 (SSH) is permitted through your firewall.
- If the announce is successful, the vSensor will appear at the *Data Sources > Network > Sensors* page.
- If **“Auto Pairing”** is enabled in *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing*

and Registration, the pairing process will begin automatically.

  - Enabling **"Auto Pairing"** is a best practice during rollout.

- If **"Auto Pairing"** is not enabled, the vSensor must be manually paired by clicking on the "Pair" icon 🔗.
  - You will then be presented with a dialog box where you can start the pairing process.
- Initially you will see the **"Pairing Status"** as **"Pairing"** once the vSensor has successfully announced itself to the Brain.
  - Once pairing is complete, the **"Pairing Status"** will change to **"Paired"**, and the **"Status"** should change to **"Forwarding"** once traffic is successfully being forwarded from the vSensor to the Brain.



Network Sensors

SENSOR NAME	TYPE	PAIRING	LOCATION/REGION	STATUS
V564dca038e5d3b7471...	DCS	Paired	–	✓ Forwarding

Last Seen: Jan 24th 2023 13:13
LUID: 3hirmfwc
Sensor IP: 172.16.12.11
Serial Number: V564dca038e5d3b7471a1e5ead689ada6
Version: 7.4.0-11-16

10 per page

Viewing 1-1 of 1

- **Please note:**
  - vSensors, like physical Sensors, will update themselves to stay current with their Brain.
  - After pairing, the vSensor will update by receiving an update from the Brain.
    - This process is automatic, and no input is required.
  - Certain vSensor CLI functions and traffic functions will become available only after the vSensor has fully updated.
  - Depending on the specific version of the vSensor, you may see errors or warnings when running CLI functions during the period of time when the vSensor is still updating.
  - "show version" can be done at the CLI to see the current version and if the vSensor is still upgrading itself from the Brain to become fully updated. Please bear in mind that state changes are not immediate and you may need to execute this command more than once to see the state change.
- The vSensor can be renamed or have its location labeled as desired by clicking on the pencil icon ✎ on the right of the vSensor.

Additional Pairing Guidance

Pairing with new or changed Brains:

- If you have a backup of your Brain and restore it to a new Brain with the same configuration (IP or hostname), previously paired Sensors (including vSensors) will connect to the new Brain automatically as the Sensor state is saved in the backup.
- If the Brain IP has changed but otherwise remains the same, the vSensors may be updated to the new IP address using the `"set brain"` command.
- Existing tunnels have to terminate to re-establish connection to a new Brain. This can be accomplished a few different ways.
  - Naturally, because the original Brain is no longer reachable due to firewall change, hardware or software failure, etc.
  - Unpairing the vSensor from the original Brain and having the vSensor attempting communication to

  the original Brain.
  - Using the "**`set brain`**" command at the CLI will terminate an existing tunnel and attempt to start pairing with a new Brain.
- If you have a Brain that will not be restored from backup that you wish to pair an existing vSensor to, this is possible via the use of the "Sensor Registration Token".
  - Retrieve or generate a current Sensor Registration Token from *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration* in the Brain GUI.
  - Perform the "**`set registration-token <token>`**" command at the Sensor CLI.
  - Finally perform the "**`set brain <IP or Hostname>`**" command at the Sensor CLI (depending on if you have selected to pair via the management IP or DNS name in *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration).*

vSensors and Pairing by Hostname vs IP

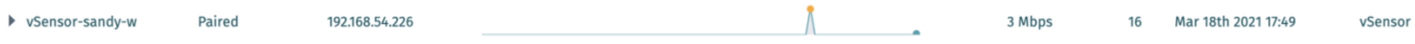
- The vSensor image downloaded from the Brain will use, by default, the Brain's IP address for pairing.
- You will need to set the *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration* **"Pair using DNS name"** option to generate the virtual machine image that points at a hostname.
- When this setting is changed, it does not affect any previously paired (either by IP or Hostname) vSensors.

Traffic Validation

Please see the following Vectra support article for recommendations on network traffic that should be examined and excluded from analysis:

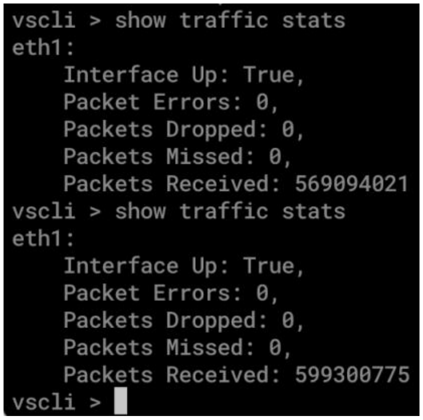
- Vectra Platform Network Traffic Recommendations

For a quick spot check to see that you are receiving any traffic at all via the vSensor you many want to check the GUI and/or CLI for statistics. If the vSensor is seeing more than 1 Mbps of traffic, this will show in the GUI under *Network Stats > Ingested Traffic* after a few minutes.



▸ vSensor-sandy-w   Paired   192.168.54.226   3 Mbps   16   Mar 18th 2021 17:49   vSensor

- You can see traffic flow immediately at the CLI of the Sensor using the "**`show traffic stats`**" command.
- Execute this command a few times in a row to see increasing packet counts.



```
vscli > show traffic stats
eth1:
    Interface Up: True,
    Packet Errors: 0,
    Packets Dropped: 0,
    Packets Missed: 0,
    Packets Received: 569094021
vscli > show traffic stats
eth1:
    Interface Up: True,
    Packet Errors: 0,
    Packets Dropped: 0,
    Packets Missed: 0,
    Packets Received: 599300775
vscli > 
```

After sending traffic to your Sensors, it is a best practice to validate that the traffic observed meets quality standards required for accurate detection and processing. Vectra's Enhanced Network Traffic Validation feature provides alarms and metrics that can be used to validate the quality of your traffic. Please see the following Vectra support article for details:

- Enhanced Network Traffic Validation (CLI)

Worldwide Support Contact Information

- Support portal: https://support.vectra.ai/
- Email: support@vectra.ai (preferred contact method)
- Additional information: https://www.vectra.ai/support