VMware Virtual Sensor (vSensor) Deployment Guide

Version: October 8, 2025

Table of Contents

Introduction

This guide is intended to help customers or partners deploy vSensors in VMware 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), vCenter integration, deployment of the vSensor in VMware, 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.

VMware 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 VMware vSensor Images

The Brain makes a VMware OVA 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 from the Brain. Once a vSensor has been deployed, it will update itself as needed, staying current with its Brain.

- Please note that as your Vectra Brain is updated, the OVA for VMware is also updated.
  - If you deploy additional VMware vSensors in the future, always download a fresh copy of the OVA from an up-to-date Brain to ensure you are working with the latest code.
  - vSensor images are retrieved from the Brain when using either the Respond UX and Quadrant UX.

VMware vSensor Resource Requirements and Performance

Resource Type	Requirement and Performance				
Performance *	500 / 250 Mbps	1 / .5 Gbps	2 / 1 Gbps	5 / 2.5 Gbps	20 / 10 Gbps
Capture Interfaces	2 **	2 **	4	4	4
CPU Cores	2	4	8 ***	16 ***	32 ***
Memory	8 GB	8 GB	16 GB	64 GB	114 GB
Storage	100 GB	150 GB	150 GB	600 GB ****	830 GB ****
VMware vSphere	6.5 to 8 (5.x was supported through version v6.14, 6.0 was supported through v6.19) *****				
vSwitch Type	VMware Virtual Standard Switch (VSS) or VMware Distributed Switch (VDS a.k.a. dvSwitch)				

- * 1st number represents NDR/Detect only performance, 2nd number represents performance with NDR/Detect and Match and/or Suspect Protocol Activity detections enabled.
- ** 2-core and 4-core vSensors can use up to 4 capture ports if the RAM is updated to at least 10GB.
- *** 2 and 4 core vSensors throttle CPU usage while 8, 16, and 32 core versions do CPU pinning to maintain performance
- **** The 16 and 32 core vSensors will need their configuration modified after deployment due to limitations in what can be preconfigured when using one image file for multiple different deployment configurations.
  - Please see: Modifying 16 and 32 core vSensors after deployment for instructions.
  - 16 core requires 600 GB storage, 32 core requires 830 GB storage and added ethernet configuration.
  - 32 core may need NUMA parameters adjusted in advanced VM configuration options.
- ***** **Special Note: Vectra supported VMware hardware versions**
  - Vectra supports only versions 11 and 15 of VMware hardware.
  - **DO NOT** update the hardware version ever (during deployments, upgrades, or in any other situation).
    - This includes updating from v11 to v15.
    - Redeployment is the only supported way to change hardware between supported versions.
  - If you move to an unsupported hardware version, Vectra support will direct you to redeploy any VMware vSensor that is running an unsupported version. Downgrades are unsupported.
- VMware based vSensors do not support DirectPath or SR-IOV passthrough.
- VMware based vSensors do not support emulated network adapters.
- VMware based vSensors do support paravirtualized NIC. Vectra uses the VMXNET3 driver for all ports.
- The virtual CPU must support a minimum SSE instruction level of 4.2 and must support the POPCNT (population count) instruction. This requires the hypervisor host to run one of the following processors or later:
  - Intel Nehalem (2008) processors and newer
  - AMD Bulldozer (2011) processors and newer
- Vectra recommends that Sensors are configured to use storage local to the hypervisor and are not stored on a SAN. Vectra vSensors require extremely high throughput from their disk storage and this throughput cannot normally be sustained by SAN systems without impact to other SAN users.
- See this support article for additional guidance around Storage/SANs, networking requirements, vMotion, Enhanced vMotion compatibility, and unsupported hypervisors.

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, please see Firewall Requirements for Vectra Appliances. VMware vSensor specific requirements are listed below:

Connectivity Requirements for VMware 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
Brain	vCenter	Configured TCP Port(s)	Physical Hosts view, vCenter Host ID input, vCenter Host context, vCenter alerts

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.
- Command Line access can also be obtained via the console in your hypervisor.

About VMware vCenter Integration

vCenter integration from the Vectra Brain for your deployment enables a number of features:

- "Virtual Infrastructure" view.
- vCenter host information artifacts help to feed Vectra's automated Host ID.
- Additional VMware context is available for analysts on VMware hosts.
- vCenter alerts are possible as an additional notification type.

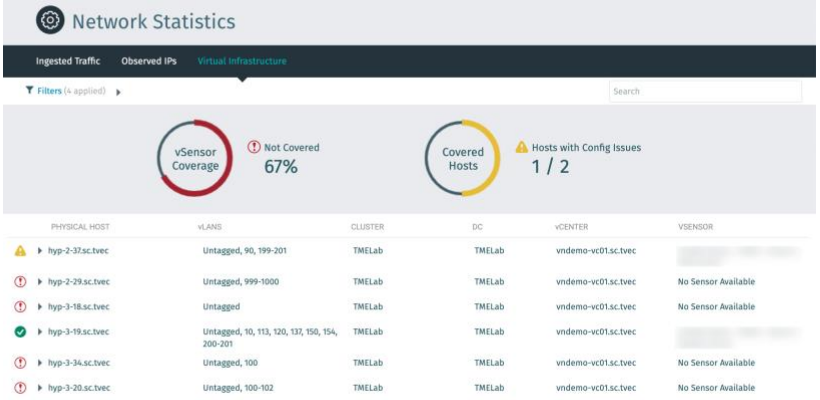
Virtual Infrastructure View

Enabling the vCenter API query connectivity helps with VMware vSensor deployment planning by identifying the physical hosts, clusters and data centers that currently have vSensor coverage, and those that do not have coverage.

Enabling the vCenter connection also shows available resources on physical VMware hosts and exposes any configuration errors that might be affecting packet capture. This view, seen in your Vectra GUI at the *Network Stats* > *Virtual Infrastructure* page, helps the Vectra admin identify the exact requirements that need to be conveyed to VMware operational teams.

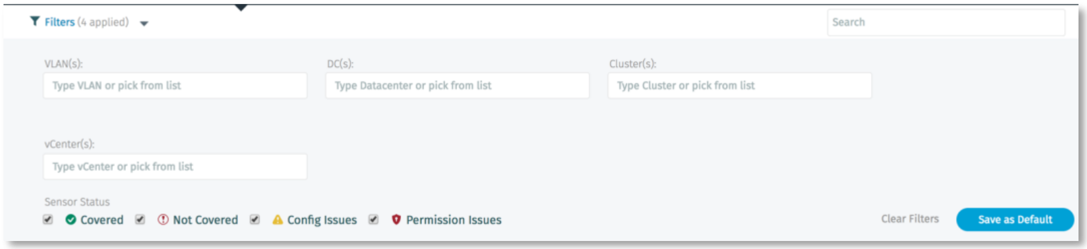
With this integration, the security team may not need to rely on the IT team to be notified of changes impacting them.

Once the vCenter integration is configured a *Network Stats* > *Virtual Infrastructure Hosts* page is enabled in the Vectra GUI:



	PHYSICAL HOST	VLANS	CLUSTER	DC	VCENTER	VSENSOR
⚠	▸ hyp-2-37.sc.tvec	Untagged, 90, 199-201	TMELab	TMELab	vndemo-vc01.sc.tvec	[redacted]
❗	▸ hyp-2-29.sc.tvec	Untagged, 999-1000	TMELab	TMELab	vndemo-vc01.sc.tvec	No Sensor Available
❗	▸ hyp-3-18.sc.tvec	Untagged	TMELab	TMELab	vndemo-vc01.sc.tvec	No Sensor Available
✓	▸ hyp-3-19.sc.tvec	Untagged, 10, 113, 120, 137, 150, 154, 200-201	TMELab	TMELab	vndemo-vc01.sc.tvec	[redacted]
❗	▸ hyp-3-34.sc.tvec	Untagged, 100	TMELab	TMELab	vndemo-vc01.sc.tvec	No Sensor Available
❗	▸ hyp-3-20.sc.tvec	Untagged, 100-102	TMELab	TMELab	vndemo-vc01.sc.tvec	No Sensor Available

The filter dropdown allows you to determine what is shown on the "Virtual Infrastructure" page:



VLAN(s): Type VLAN or pick from list

DC(s): Type Datacenter or pick from list

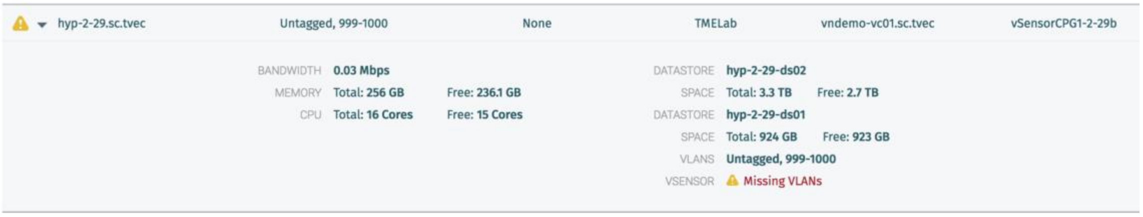
Cluster(s): Type Cluster or pick from list

vCenter(s): Type vCenter or pick from list

Sensor Status

☑ ✓ Covered ☑ ❗ Not Covered ☑ ⚠ Config Issues ☑ Permission Issues

A red exclamation point means that a particular physical hypervisor is NOT covered. This either means that there is no vSensor installed on the hypervisor or that the installed vSensor cannot be detected. A yellow warning sign icon means that there is a configuration issue with the installed vSensor:

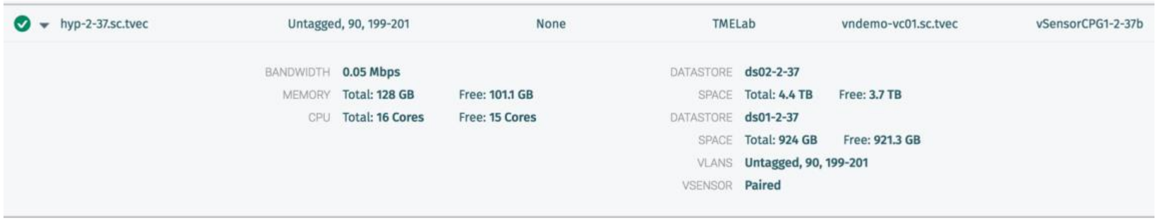


⚠ ▾ hyp-2-29.sc.tvec | Untagged, 999-1000 | None | TMELab | vndemo-vc01.sc.tvec | vSensorCPG1-2-29b

BANDWIDTH 0.03 Mbps
MEMORY Total: 256 GB Free: 236.1 GB
CPU Total: 16 Cores Free: 15 Cores

DATASTORE hyp-2-29-ds02
SPACE Total: 3.3 TB Free: 2.7 TB
DATASTORE hyp-2-29-ds01
SPACE Total: 924 GB Free: 923 GB
VLANS Untagged, 999-1000
VSENSOR ⚠ Missing VLANs

A green checkmark means that the vSensor is configured and functioning properly:



✓ ▾ hyp-2-37.sc.tvec | Untagged, 90, 199-201 | None | TMELab | vndemo-vc01.sc.tvec | vSensorCPG1-2-37b

BANDWIDTH 0.05 Mbps
MEMORY Total: 128 GB Free: 101.1 GB
CPU Total: 16 Cores Free: 15 Cores

DATASTORE ds02-2-37
SPACE Total: 4.4 TB Free: 3.7 TB
DATASTORE ds01-2-37
SPACE Total: 924 GB Free: 921.3 GB
VLANS Untagged, 90, 199-201
VSENSOR Paired

vCenter Host ID artifacts

Vectra's automated Host ID is a key benefit for analysts using the system. The goal of Vectra's Host naming is to provide human-readable names associated with known Hosts.

Host names result from known information about the Host. Each observed name is referred to as an "artifact". Artifacts will typically be added to a Host record over time as more Host activity is seen and better associations are made. Host artifacts may be removed from a Host depending on the observed behaviors.

Hosts are tracked internally in a name agnostic manner. When assessing Host naming in your deployment, it is important to understand that Host names are decided at the time of viewing the web page. It is therefore expected that displayed Host names will change over time to reflect the most human readable name given the artifacts available at the time of page display.

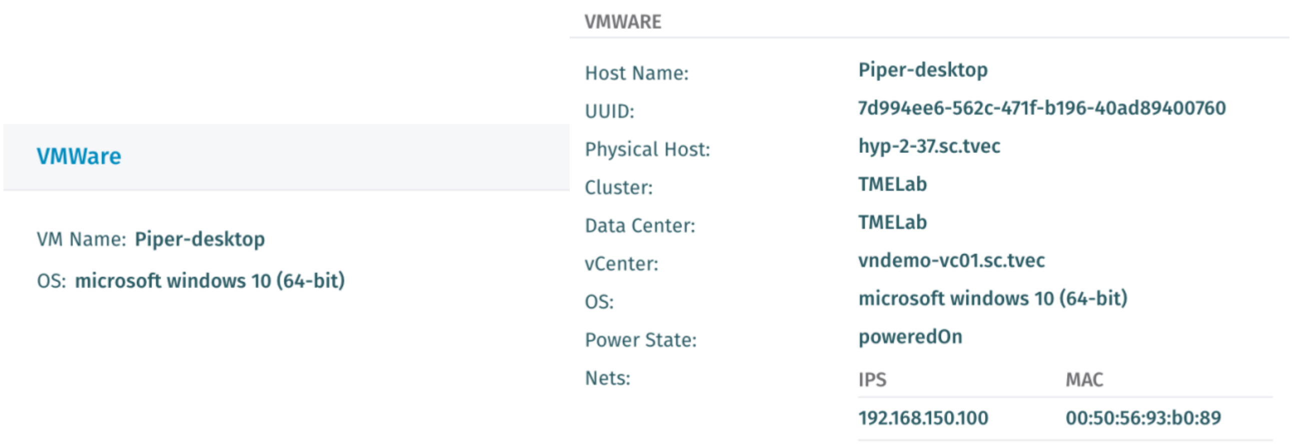
The hostname obtained through vCenter/vSphere integration via an active query using the vCenter API is a key artifact when available in a customer environment. It is considered a best practice to enable the vCenter integration even if you will not deploy VMware vSensors in your environment.

For additional information regarding Vectra's automated Host ID, please see the following Vectra support portal article:

Understanding Vectra NDR Host Naming

Additional Host context for analysts

When an analyst views a Host that is running VMware tools and reporting back to vCenter/vShere with the Vectra vCenter integration enabled, additional context about the Host is available. To view this context, look at the left-hand side of the Host's page in the Vectra UI for summary information including the VM name and operating system as reported by the vCenter API. The **"Host Details"** view has a more complete view. Below are examples of each:



VMWare

VM Name: **Piper-desktop**

OS: **microsoft windows 10 (64-bit)**

VMWARE

Host Name:	Piper-desktop	
UUID:	7d994ee6-562c-471f-b196-40ad89400760	
Physical Host:	hyp-2-37.sc.tvec	
Cluster:	TMELab	
Data Center:	TMELab	
vCenter:	vndemo-vc01.sc.tvec	
OS:	microsoft windows 10 (64-bit)	
Power State:	poweredOn	
Nets:	IPS	MAC
	192.168.150.100	00:50:56:93:b0:89

vCenter alerts

Once the vCenter integration is configured, additional alerts are available that are specific to changes in the environment that may merit security consideration. To enable these alerts navigate to *Settings* > *Notifications* > *Alert Emails*, select the **"Edit"** or pencil icon, scroll to the bottom of the Alert Emails settings, and enable the toggle to **"Send vCenter alerts"**. Some example scenarios where an alert will be sent are:

- A new physical hypervisor where a vSensor may be needed has been added to the environment.
- A vSensor has been moved or powered down.

A VM is moved from a Host that is monitored by a vSensor to a Host that is not monitored by a vSensor.

Enabling vCenter Integration

Prepare vSphere account for Brain access

To connect the Brain to vSphere, a vSphere user account and password must be configured into the Brain. The vSphere user account must have at least global, read-only rights. The Brain will not attempt to write any data to your VMware environment.

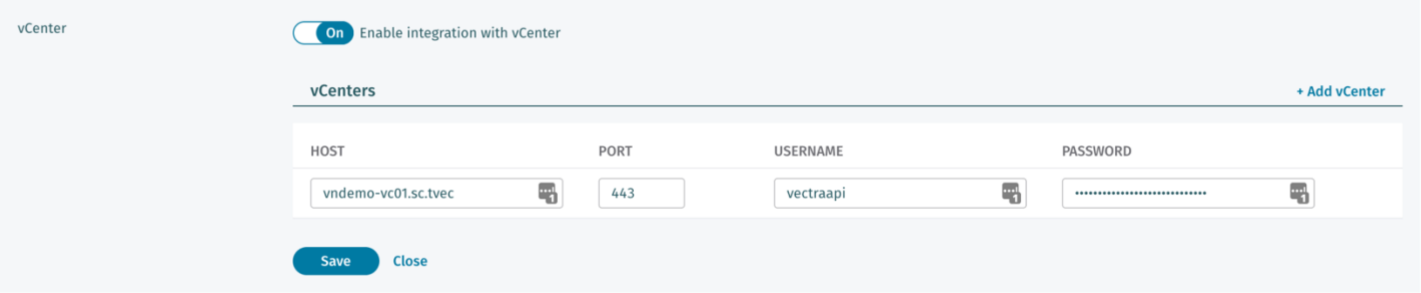
To ensure that the vSphere user/group the Brain will use has global, read-only access, use the following steps in the vSphere UI:

- From the vSphere Administration page select *Access* > *Global Permissions*
- Click the **plus** sign to display the global permissions dialog
- At the bottom of the left pane, click **Add**
- Ensure the domain is set to the proper domain, select the **users** or **groups** you intend to use in Vectra's configuration to connect to vCenter's API and click **OK**
- In the **Assign Role** section, select **Read-Only** from the drop-down list
- Make sure the **Propagate to children** checkbox is selected, and click OK

Configure vCenter/vSphere integration

You will need to have the IP or hostname of your VMware vCenter/vSphere server. You can configure multiple integrations if you have more than one server to connect to. You will also need to have the port number, username and password.

Navigate to *Settings* > *External Connectors* > *vCenter* and edit the vCenter settings. Any previously configured vCenters will be shown in this area:



vCenter

On Enable integration with vCenter

vCenters + Add vCenter

HOST	PORT	USERNAME	PASSWORD
vndemo-vc01.sc.tvec	443	vectraapi	••••••••••••••••••••••••••••••

Click on the + Add vCenter to add an additional vCenter, fill in the blanks, and click **"Save"**

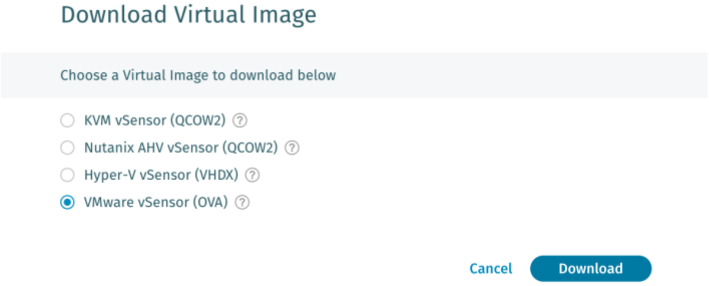
vSensor Deployment in VMware

Requirements

- IP address and subnet mask for the Management interface of the vSensor.
- DNS server addresses.
- Administrative access to you VMware vCenter/vSphere console or authorization to deploy via API connection.
- To configure your vSensor after the initial deployment, you will need access to the vSensor Command Line Interface (CLI) either via the console in your hypervisor or via SSH.
  - The vSensor can be deployed with a static IP when deployed using the vSphere UI in vCenter. When deployed using the embedded host client in ESXi, only DHCP is available for initial deployment.
  - You must know the IP assigned via DHCP to SSH to the CLI, otherwise use the hypervisor console.
- VMware specific information is required.
  - vSphere hostname or IP, vm name for the vSensor, datacenter hostname or IP, vmhost to deploy on, datastore, management portgroup, capture portgroup, vswitch, # of cores.
- Optional VMware specific information for vSphere api based deployment from the Brain CLI.
  - vSphere port, resource path, hostname to assign, username, password.
- Only use supported VMware hardware versions (v11 or v15). See earlier guidance for more details.
- For production monitoring, ensure that the vSensor VM is kept running 24x7, and ensure that the hypervisor does not overcommit resources or otherwise misrepresent the resources it is providing to the vSensor.
- vMotion should not be enabled for vSensors.

Downloading the latest vSensor VMware OVA image

The current vSensor OVA image for deployment can be downloaded from the Brain by clicking the blue "Download Virtual Image" link at the top right of the *Data Sources* > *Network* > *Sensors* page in your Brain and then selecting the VMware vSensor (OVA) option. It can take up to 45 minutes for newly deployed Brains to have all images fully processed and available for download. If they don't show as available yet, please try again later.



Download Virtual Image

Choose a Virtual Image to download below

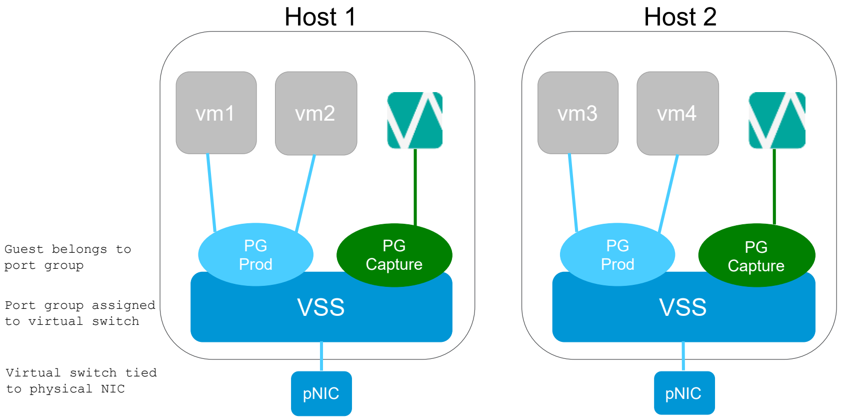
○ KVM vSensor (QCOW2)
○ Nutanix AHV vSensor (QCOW2)
○ Hyper-V vSensor (VHDX)
● VMware vSensor (OVA)

VMware vSwitch types and port group guidance

VMware has two different types of virtual switches, VSS and VDS (VMware/vSphere Standard Switch and VMware/vSphere Distributed Switch). VSS is available with any license level (even "free" ESXi) while VDS is a feature that is only available with the Enterprise Plus license level.

vSensors need to have port groups that are configured in promiscuous mode to allow analysis of all desired traffic on the physical hypervisor (ESXi host) they are deployed on. VLANs can be used to limit what traffic is analyzed by the vSensor as well. The following diagrams provide some additional detail:

VSS (VMware/vSphere Standard Switch):

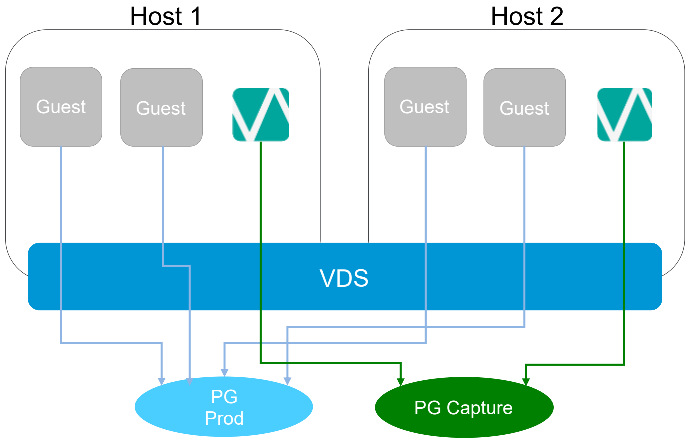

Host 1
Host 2
vm1
vm2
vm3
vm4
Guest belongs to port group
PG Prod
PG Capture
Port group assigned to virtual switch
VSS
Virtual switch tied to physical NIC
pNIC

Promiscuous Mode – If set, all traffic on VSS matching VLAN filter sent to port group

VSS vSwitch - Defined per physical host
Port Group - Defined per VSS

PG Capture
- Set Promiscuous Mode
- VLAN Filter: 0 (untagged), 4095 (all), or VLAN-ID

VDS (VMware/vSphere Distributed Switch):



Promiscuous Mode- If set, all host traffic on VDS matching VLAN filter sent to port group of that host

VDS vSwitch - Spans ≤ 350 physical hosts
Port Group - Defined per VDS

PG Capture
- Set Promiscuous Mode
- VLAN Filter: 0 (untagged), 0-4094 (all), VLAN ranges, comma separated

Preparing Port Groups

- Capture Port Groups
  - VSS
    - Security – Promiscuous mode must be set to Accept to ensure that the vSensor is able to receive all packets.
    - Properties – It is a best practice to set the VLAN ID set to All (4095) to ensure that no packets are filtered by VMware before reaching the vSensor.
  - VDS
    - Security – Promiscuous mode must be set to Accept to receive all packets.
    - VLAN – VLAN type must be set to VLAN trunking with the VLAN range set to 0-4094 as a best practice to ensure that no packets are filtered by VMware before reaching the virtual Sensor. Limit VLANS if required.
    - Forged transmits and MAC address changes should be set to "Accept".
  - VLANs can be limited by the customer to eliminate certain VLANs from having their traffic analyzed.
    - An example would be VLANS dedicated to I/O traffic such as iSCSI or vMotion.
    - This can reduce load on the vSensor and allow smaller vSensors to be deployed.
- Management Port Groups
  - These may already exist in the customer environment.
  - This is what needs to be able to be accessed from administrator workstations to log in to the Vectra Brain UI and CLI.

VMware physical hosts and vSensor coverage

When deploying vSensors with VSS, it is clear that customers need a vSensor per physical hypervisor or standalone ESXi host because the VSS is unique to each physical host. In a VDS scenario, it is still required to deploy a vSensor per physical host. VDS allows a single distributed switch to be shared across physical hypervisors in your environment, but local traffic (per hypervisor) is not forwarded across the VDS by default even when another host has a port group set to promiscuous mode in the VDS. The recommendation is as follows:

- Create a port group for monitoring (e.g. "Monitor"). You need only create one such port group and it will be available for the entire VDS
- Place one vSensor per physical host
- Place the capture interface from the vSensors into the "Monitor" port group
- Set promiscuous mode for "Monitor" to "Accept" and set the VLAN for Monitor to "0-4094" in order to monitor all current and future port groups / VLANs that may be placed on any host in the VDS
  - Alternately you could use specific VLANs (singles and/or ranges) if you only ever wanted a subset of the traffic, or if you wanted to exclude certain VLANs (as you would want to do for I/O VLANs, e.g. those dedicated to iSCSI, FCoE, vMotion, etc.)
- This will send any traffic that goes over the local (within the same ESXi host) vSwitch instance, matching the VLANs specified, to the local Monitor port group instance, to be picked up by the local vSensor's capture I/F. The traffic will not forward across the VDS.

VMware remote monitoring solutions, e.g. remote packet mirroring, should not be used to replicate traffic from one hypervisor to a vSensor running on another hypervisor.

It is best practice to think of vSensors as being tied to the physical host they are deployed on. They should generally not participate in vMotion or clustering/failover configurations. In situations where you are not fully deploying vSensors to have full coverage of your virtual environment (i.e. a limited deployment PoV), customers have successfully used affinity rules to keep a vSensor with a workload (AD Server) and participate in vMotion. They also used anti-affinity rules to keep from having more than 1 vSensor on the same physical host. These cases should be the exception, rather than the rule, and are not suitable for production deployments. Please work with your Vectra sales/deployment/support teams for additional guidance.

VMware networking interface guidance

- Management interface (MGT1)
  - This is referred to in the vSphere UI as mgt1 when creating the VM from the OVF template.
  - After the VM has been created it is represented as "Network adaptor 1" in VMware.
- Dedicated capture ports
  - These are referred to in the vSphere UI as eth0 when creating the VM from the OVF template.
  - After the VM has been created they are represented as "Network adaptor 2", etc in VMware.

Deploying the OVA

There are a number of ways to deploy the OVA into your VMWare environment:

- Using your vCenter/vSphere client or embedded host client for standalone ESXi servers.
- Using the "`provision vmware vsensor`" CLI command from the Brain.

Deploying using your vCenter/vSphere client or embedded host client for standalone ESXi servers

vCenter/vSphere versions and clients vary. The general process for deployment via your vCenter/vSphere client will also differ slightly if you use VSS or VDS vSwitch types. You may need to make adjustments for your environment.

For example, the web UI for ESXi 6.5 not have full feature support for some standard OVA features including specifying **"deployment options"** within an OVA. Due to this it is not possible to deploy vSensors on standalone ESXi 6.5 Update 1 using the web UI. In these cases, it is recommended to deploy using the vCenter app or vSphere client. The Vectra CLI command "`provision vmware vsensor`" is another option. For more information see this article:

- Provision vSensor using VCLI

The general process for using the vCenter application or vSphere client to deploy the OVF is as follows:

- Login and navigate to Hosts and Clusters.
- Right click on the host where the vSensor will be deployed and select **"Deploy OVF Template".**
- Make sure to only use supported VMware hardware versions (see earlier guidance for details).
- A pop up will walk you through configuring the virtual appliance.
  - Browse to the OVA that was downloaded from the Vectra Brain.
  - Give the vSensor VM a name (this will appear in vSphere and in the Brain).
  - Also assign it to a datacenter, cluster, and folder appropriate for your environment.
  - Select the datastore to house the virtual environment.
  - Please do not make changes to virtual disk format (it should be thick provisioned).
  - The port groups used for MGT1 and for capture should have already been configured as per the previous guidance above for **"Preparing Port Groups".**
  - Assign Network Adaptor 1 (MGT1) to the port group used to manage the vSensor.
  - Assign Network Adaptor 2 to the capture port group.
  - Configure DHCP or static assignment for MGT1 (only available in vSphere/vCenter UI).
    - If a static IPv6 address is assigned during deployment, IPv6 support will be automatically enabled. Please see IPv6 Management Support for Vectra Appliances for more details.
- Additional NICs for capture can be added after deployment if required and you follow the VMware vSensor Resource Requirements and Performance guidance from earlier in the doc.
- Review the final details and **DO NOT** enable **"Power on Upon Completion"** if you need to add configuration options for the 32 core VM and/or change the disk size for 16 and 32 core VMs.
  - Please see: Modifying 16 and 32 core vSensors after deployment for instructions.
- If you do deploy using the embedded host client on a standalone ESXi host, you may receive a warning about ignoring a disk, but this message can be ignored. DHCP is the only option for initial deployment when using the embedded host client for ESXi.

Deploying a vSensor using the Vectra CLI on your Brain

The CLI tool is an easy and convenient way to deploy on vCenter/vSphere and ESXi standalone servers. To deploy using this method, follow these steps:

- Ensure the capture portgroup and management port groups are already created as per the previous guidance above for **"Preparing Port Groups".**
- Ensure any firewall allows the Brain to connect to the vCenter server (if applicable) and the ESX/vSphere server on port 443 (or alternate port if configured).
- Login to VCLI on the Brain using the "**`vectra`**" user.
- Run the "**`provision vmwware vsensor`**" command using the appropriate options.
- Once successfully deployed and powered on, the new vSensor should automatically pair to the Brain if Automatic Pairing is enabled under *Data Sources* > *Sensors* > *Sensor Configuration* > *Sensor Pairing and Registration* on your Brain.

Options for the "**provision vmware vsensor"** command can be displayed at the Brain CLI as below:

```
vscli > provision vmware vsensor -h
Usage: provision vmware vsensor [OPTIONS]

Uses ovftool along with the supplied information to provision new virtual
 sensors to vCenter or a standalone ESXi hypervisor.

Options:
 -vs, --vsphere TEXT IP or hostname of vCenter/vSphere instance [required]
 -vm, --vmname TEXT Virtual machine name to assign to the vSensor [required]
 -ds, --datastore TEXT Name of the datastore to create the virtual machine on [required]
 -m, -mp, --mgmt_pg TEXT Management NIC's portgroup name [required]
 -cp, --capture_pg TEXT Capture NIC's portgroup name [required]
 -s, -vsw, --vswitch TEXT Name of the vSwitch that the capture portgroup is on [required]
 -dc, --datacenter TEXT Name of the data center where the vsensor will be created on (vCenter only)
 -vh, --vmhost TEXT Name of the physical host that the vSensor will be created on (vCenter only)
 -d, --dhcp Select DHCP or static IP, Netmask, Gateway for vSensor management (only supported on vCenter)
 -mip, --mgmt_ip TEXT Static Management IP address (only supported on vCenter)
 -mnm, --mgmt_netmask TEXT Static Management IP netmask (only supported on vCenter)
 -mgw, --mgmt_gw TEXT Static Management gateway IP address (only supported on vCenter)
 -n, --dns TEXT Comma separated list of DNS server IP addresses (only supported on vCenter)
 -c, --cores [2|4|8|16] Number of cores for vSensor to use (default 4)
 -p, --port INTEGER vSphere port (default is 443)
 -r, -rp, --resource_path TEXT Folder/resource path in which a host is located, e.g. "Folder Name/Cluster name"
(vCenter only)
 -f, -fp, --force_promiscuous if provided, promiscuous mode will be enabled on capture portgroup automatically
 -hn, --hostname TEXT vSensor hostname to assign (only supported on vCenter)
 -u, --username TEXT vCenter/vSphere username (you will be prompted if not provided)
 -pw, --password TEXT vCenter/vSphere password (you will be prompted if not provided)
 --wait-for-ip If selected, command returns only when the sensor successfully got an IP address
 -h, --help Show this message and exit.
```

Command syntax:

```
provision vmware vsensor < -vs vsphere > < -vm vmname > < -ds datastore > < -m mgmt_pg > < -cp capture_pg > < -s
vswitch > [ -dc datacenter ] [ -vh vmhost ] [ -d ] [ -mip mgmt_ip ] [ -mnm mgmt_netmask ] [ -mgw mgmt_gw ] [ -n
dns ] [ -c cores( 2 | 4 | 8 | 16 ) ] [ -p port ] [ -r resource_path ] [ -f ] [ -hn hostname ] [ -u username ] [ -
pw password ] [ --wait-for-ip ]
```

Example command:

```
provision vmware vsensor -vs "vsphere.local" -vm "vSensor-01" -ds "esxhost2 NVMe" -m "10x3 Management Network" -cp
"Vectra Analyzer" -s vSwitch1 -dc "Oakland" -vh "Production 17" -mip 10.0.3.92 -mnm 255.255.255.0 -mgw 10.0.3.1 -n
10.0.6.10 -c 2
```

Please note:

- The Vectra provision command uses VMware's ovftool along with the supplied information to provision vSensors.
- Not all arguments are required.
  - For example, if a username or password is not specified, you will be prompted for them.
  - If a number of cores is not specified, the default of 4 will be used.
  - For additional information and troubleshooting if this command fails, please see the Provision vSensor using VCLI Vectra support portal article.

Special Note: Embryo state of vSensor before pairing and updating

Immediately after the initial deployment of a vSensor, it is in what is known as an "embryo" state. The vSensor needs to be paired to a Brain, then receive a software update from the Brain, and finally update to become fully functional. During the time before pairing and updating, not all vSensor commands are functional yet.

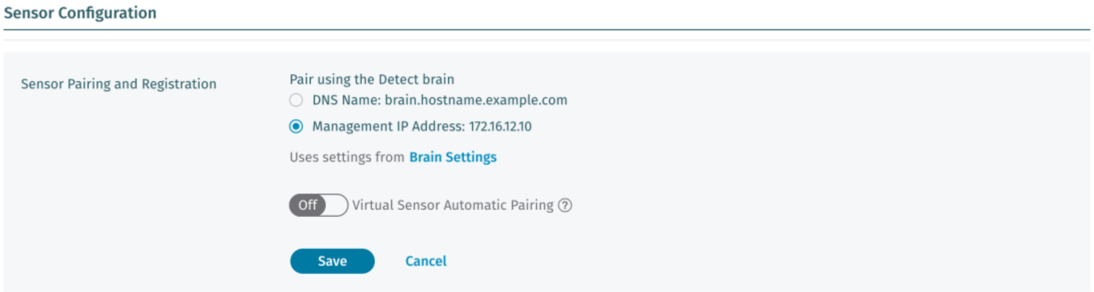
To login to a vSensor, the username is "**vectra**" and the initial password is "**changethispassword**". After the vSensor is paired and updated, the initial login to the updated vSensor will force a password change.

For example, the "**show traffic stats**" command does not exist on vSensors that are in embryo state. To determine if your vSensor is still in embryo state, you can use the "**show version**" command. If the version string is empty, then the vSensor is still in embryo state. After a vSensor has been paired, upgrading will become "True" after the vSensor has successfully downloaded an update image from the Brain and has begun updating. Please see the example below for what a vSensor will output when in embryo state:

```
vscli > show version
Upgrading: False
Version:
```

While in embryo state, it is recommended to only use commands related to pairing such as the following:

- "**set brain**"
  - Ordinarily, this command is not necessary as the vSensor image that was downloaded from your Brain is already set to pair with the Brain by hostname or by IP address, depending on how *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration* is configured. As an example, in the below screenshot, the IP address of the Brain would be embedded into the vSensor image that was downloaded from the Brain so that it already knows where to attempt to pair when it is booted.



Sensor Configuration

Sensor Pairing and Registration

Pair using the Detect brain

○ DNS Name: brain.hostname.example.com

● Management IP Address: 172.16.12.10

Uses settings from Brain Settings

Off Virtual Sensor Automatic Pairing

  - For DNS Name to be an option, a hostname must be configured for your Brain in *Data Sources > Network > Brain Setup > Brain*. If no hostname is configured, then "Management IP Address" will be the only option available.
  - It is recommended to configure a hostname and use it for pairing when possible. Doing so typically makes failover situations easier to manage when IPs of Brains may change in failover scenarios.

- “`set registration-token`”
  - Typically, this is also not required for embryo vSensors as the downloaded vSensor is already configured to be able to pair with the Brain it was downloaded from.
  - If you need to pair with a different Brain, the “`set registration-token`” command will enable the vSensor to pair with a Brain that did not provide the initial vSensor image download.
  - Sensor registration tokens are created on a Brain and are good for 24 hours after creation. If you need to generate one, navigate to *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration.*

Modifying 16 and 32 core vSensors after deployment

As per the VMware vSensor Resource Requirements and Performance section, the 16 and 32 core vSensor configurations will need to be modified after deployment due to limitations in what can be preconfigured in the image that is shared for multiple vSensor configurations.

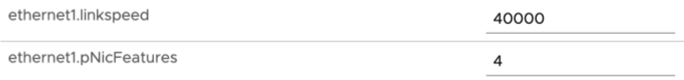
Ideally these modifications will be done before the Sensor is powered on for the first time. It is still possible to make the required modifications if the Sensor was powered on previously, but the overall process is simpler and requires less manipulation if you do this before the Sensor is power on. In any case the Sensor must be shut down before making the changes and then powered on after saving the changes.

To modify the disk size for the 16 or 32 core vSensor:

- Shut down the vSensor if it is already powered on.
- Edit the settings in VMware (embedded ESXi client or vCenter/vSphere client) for the Sensor.
- Change the disk size to:
  - 600 GB for the 16 core vSensor.
  - 830 GB for the 32 core vSensor.
- Save the configuration and power on the vSensor.

32 Core vSensor Ethernet Modification:

- This is only required for the 32 core vSensor.
- Shut down the vSensor if it is already powered on.
- Edit the settings in VMware (embedded ESXi client or vCenter/vSphere client) for the Sensor.
- Go to *VM Options > Advanced* and then edit the configuration parameters and add two new parameters:
  - If the link speed will be 10 Gbps, the linkspeed parameter is NOT required but is a best practice.
  - The default link speed is 10 Gbps for a capture NIC, modify this as required for your deployment.
    - 20 Gbps is the max throughput for the Sensor but the link speed can be set to the same as the physical NIC associated with this interface if capturing physical traffic or the aggregate bandwidth required if combining multiple sources into one feed.
  - Examples for 1st capture port (Network Adapter 2) (MGT is Network Adapter 1 / eth0):
    - Name/Key: **`ethernet1.pNicFeatures`**
      - Value: **`4`**
    - Name/Key: **`ethernet1.linkspeed`**
      - Value: **`40000`** – This represents a 40 Gbps link speed, adjust as needed for your required link speed.



ethernet1.linkspeed	40000
ethernet1.pNicFeatures	4

    - Repeat adding both parameters for any additional capture NICs (max of 4 capture NICs)
      - Use ethernet2 for 2nd capture port, ethernet 3 for 3rd capture port, etc

32 Core vSensor NUMA Configuration:

This section applies to the 32 core vSensor only. No changes are required for other vSensor sizes.

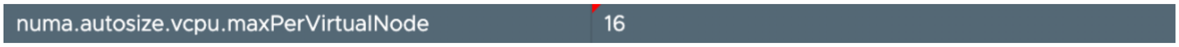
VMware provides guidance for Using NUMA Systems with ESXi in the linked documentation. Virtual NUMA Controls documents the parameters. The numa.vcpu.maxPerVirtualNode parameter controls NUMA configuration for Vectra VMware VMs. Vectra cannot set this parameter at the .ova level and on some 32 core VMware vSensors (this varies by the underlying hardware platforms used), the parameter must be set by the customer after the VM deployment or errors will be seen during boot of the VM.

If the VM reboots frequently, every 3 to 4 min, if you can see the output of "show system-health" at the CLI of your VMware vSensor and there is a message about NUMA, then you know this is the issue. To avoid the issue, it is best to check for the proper setting of the parameter before powering on the VM, and set it if required.

numa.autosize.vcpu.maxPerVirtualNode is an advanced parameter in VMware vSphere/ESXi. It controls how many vCPUs ESXi can automatically assign to a NUMA node when it is handling wide VMs. By default, ESXi sets and manages this internally based on host NUMA topology, VM sizing, and hypervisor defaults. The value of {{numa.autosize.vcpu.maxPerVirtualNode}} should be set to 16, so that each NUMA node can get an equal number of vCPUs.

To check the parameter and set it if required:

- Go to *VM Options* > *Advanced* and then edit the configuration parameters and find:



numa.autosize.vcpu.maxPerVirtualNode 16

- If the setting is 16, you are done and can close the parameters/VM options.
- If the setting is not 16, change it to 16 and save the configuration and power on the vSensor.

Capturing Physical Network Traffic Using a vSensor

It may be desirable to mirror traffic from a physical switch to a Vectra vSensor. There are two ways to mirror traffic from a physical switch into a VMware ESXi hypervisor host for monitoring by a Vectra vSensor.

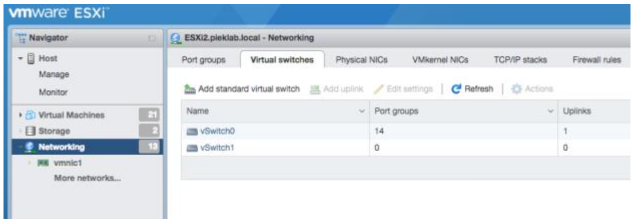
The first method utilizes a dedicated physical NIC on the host chassis to carry tagged or untagged traffic from the mirror session on the switch to the vSensor on the host. The second method utilizes a VLAN that is trunked over a link to the host.

Method 1: Dedicated link to ESXi host

Utilizing a dedicated link from the physical switch to the ESXi host may require the addition of a dedicated vSwitch due to VLAN tagging. The following procedure outlines the necessary steps required to setup ESXi's network to accomplish this.

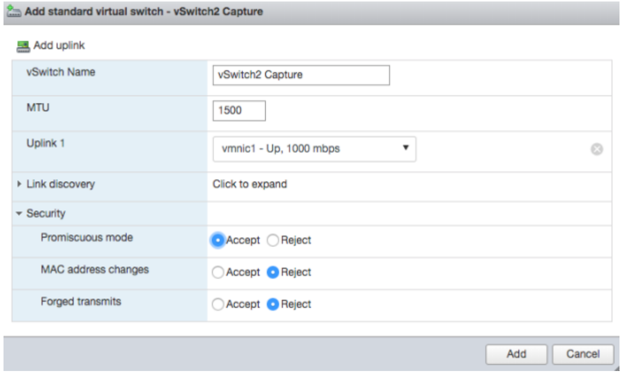
Step 1: Add a new virtual switch

- To add new virtual switch, enter the "Networking" menu and choose "Add standard virtual switch":



Name	Port groups	Uplinks
vSwitch0	14	1
vSwitch1	0	0

- Create the new vSwitch by choosing the appropriate physical NIC that is attached to the mirror output port as the Uplink.
- Under Security settings, enable "Promiscuous mode"

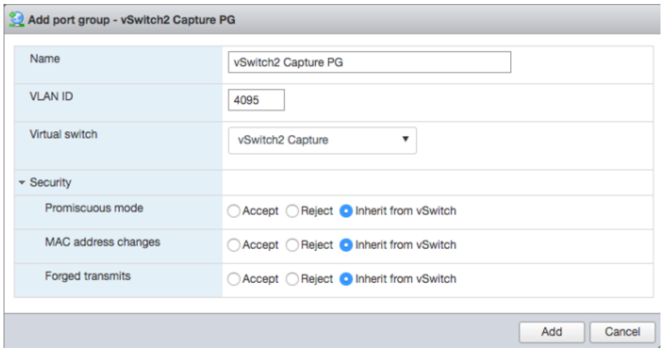


Add standard virtual switch - vSwitch2 Capture

vSwitch Name	vSwitch2 Capture
MTU	1500
Uplink 1	vmnic1 - Up, 1000 mbps
▸ Link discovery	Click to expand
▾ Security	
Promiscuous mode	● Accept ○ Reject
MAC address changes	○ Accept ● Reject
Forged transmits	○ Accept ● Reject

Step 2: Create port group for capture interface

- On the "Port groups" tab, click "Add port group":



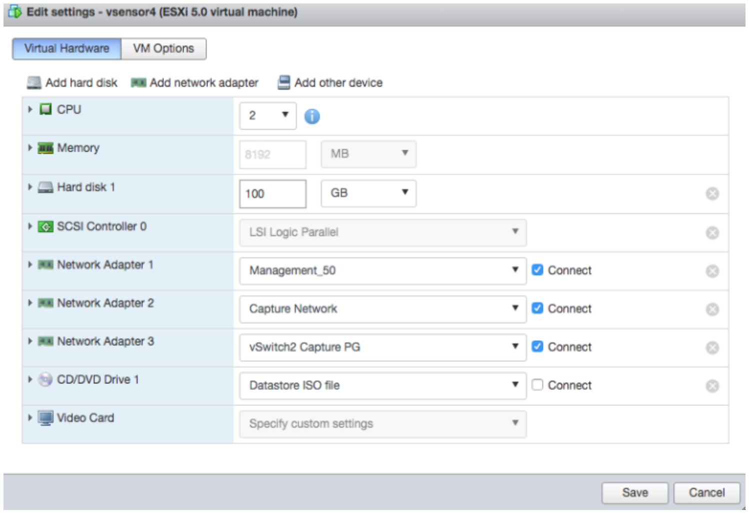
Add port group - vSwitch2 Capture PG

Name	vSwitch2 Capture PG
VLAN ID	4095
Virtual switch	vSwitch2 Capture
▾ Security	
Promiscuous mode	○ Accept ○ Reject ● Inherit from vSwitch
MAC address changes	○ Accept ○ Reject ● Inherit from vSwitch
Forged transmits	○ Accept ○ Reject ● Inherit from vSwitch

- Enter VLAN ID 4095 to monitor all VLANs being trunked (including native) over the physical link from the switch
- Select the virtual switch created in the previous step for **"Virtual switch"**
- Ensure that the port group's security settings are being inherited from the vSwitch

Step 3: Configure vSensor’s Network Adapter

- Edit the settings of the Vectra vSensor
- Select the newly created port group in the previous step for the appropriate capture interface:



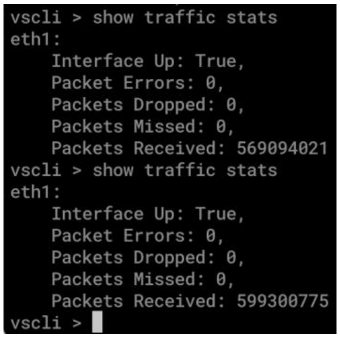
Edit settings - vsensor4 (ESXi 5.0 virtual machine)

▸ CPU	2	
▸ Memory	8192 MB	
▸ Hard disk 1	100 GB	
▸ SCSI Controller 0	LSI Logic Parallel	
▸ Network Adapter 1	Management_50	☑ Connect
▸ Network Adapter 2	Capture Network	☑ Connect
▸ Network Adapter 3	vSwitch2 Capture PG	☑ Connect
▸ CD/DVD Drive 1	Datastore ISO file	☐ Connect
▸ Video Card	Specify custom settings	

- Click **"Save"**

Step 4: Verify vSensor is receiving packets

- Log in to the vSensor's CLI
- Run the command **"show traffic stats"** several times to verify the interface is receiving traffic as expected and **"packets_received"** counts are increasing. Please note that this command will only function after the vSensor has been paired and updated from the Brain. For details, please see the earlier guidance about the embryo state of vSensors immediately after initial deployment.



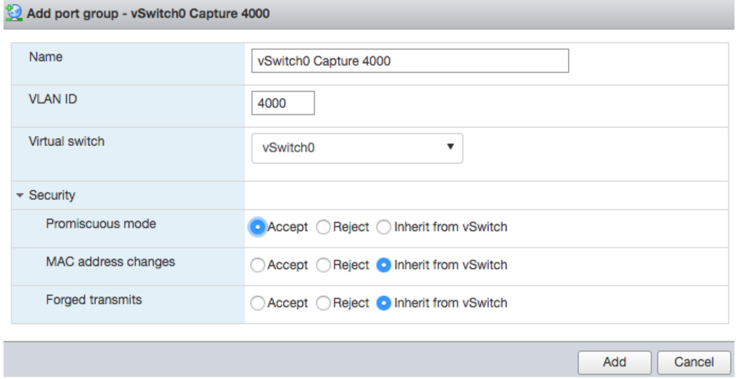
```
vscli > show traffic stats
eth1:
    Interface Up: True,
    Packet Errors: 0,
    Packets Dropped: 0,
    Packets Missed: 0,
    Packets Received: 569094021
vscli > show traffic stats
eth1:
    Interface Up: True,
    Packet Errors: 0,
    Packets Dropped: 0,
    Packets Missed: 0,
    Packets Received: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

Method 2: Utilizing a VLAN tag over an existing trunked link

When a dedicated physical link between the switch and the ESXi host is not desired or possible, a switch's mirroring session output can usually be configured to output on a VLAN. Configuration on the physical network will vary by deployment and network vendor. Please work with your networking team and/or vendor to complete physical network configuration.

Step 1: Create the port group

- Create a port group for the vSensor's capture interface
- In ESXi's Networking menu, choose the **"Port groups"** tab
- Click on **"Add port group"**
- Enter the VLAN ID that the switch will be mirroring traffic over (4000 in this case)
- Choose the appropriate virtual switch that has the physical link trunking the VLAN
- Ensure **"Promiscuous mode"** is enabled under **"Security"**

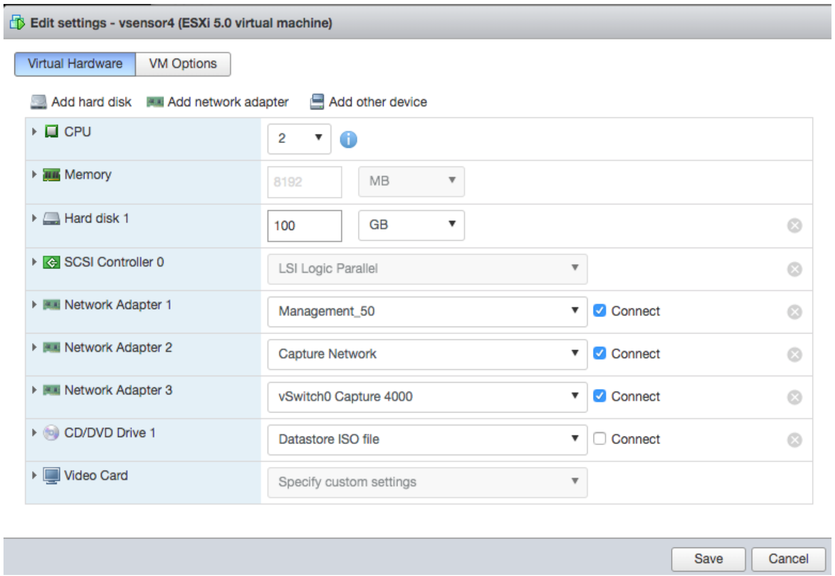


Add port group - vSwitch0 Capture 4000

Name	vSwitch0 Capture 4000
VLAN ID	4000
Virtual switch	vSwitch0 ▾
▾ Security	
Promiscuous mode	● Accept ○ Reject ○ Inherit from vSwitch
MAC address changes	○ Accept ○ Reject ● Inherit from vSwitch
Forged transmits	○ Accept ○ Reject ● Inherit from vSwitch

Step 2: Configure the vSensor

- Edit the settings of the vSensor
- Select the newly created port group in the previous step for the appropriate capture interface



Edit settings - vsensor4 (ESXi 5.0 virtual machine)

CPU: 2
Memory: 8192 MB
Hard disk 1: 100 GB
SCSI Controller 0: LSI Logic Parallel
Network Adapter 1: Management_50 ☑ Connect
Network Adapter 2: Capture Network ☑ Connect
Network Adapter 3: vSwitch0 Capture 4000 ☑ Connect
CD/DVD Drive 1: Datastore ISO file □ Connect
Video Card: Specify custom settings

Step 3: Verify vSensor is receiving packets

- Log in to the vSensor's CLI
- Run the command "**`show traffic stats`**" several times to verify the interface is receiving traffic as expected and "**`packets_received`**" counts are increasing. Please note that this command will only function after the vSensor has been paired and updated from the Brain. For details, please see the earlier guidance about the embryo state of vSensors immediately after initial deployment.

Initial vSensor Configuration at CLI

If you are not using DHCP or would like to set a static address, you will need to login to the CLI of the vSensor to set a static interface assignment. DNS for vSensors must also be configured at the CLI.

vSensor login at the CLI is very similar to logging in to physical Sensors. The primary difference is that there is no physical serial console, IPMI/iDRAC, or other ports to log in to. Logging in can be done via your hypervisor console function or using SSH to the management port if it was preconfigured with DHCP.

- 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 the Brain UI then you should be able to see its IP address in *Data Sources* > *Network* > *Sensors* as well.
  - The initial password is "**`changethispassword`**". You will be asked to change this when logging in to a paired and updated vSensor. Please see details for vSensors in embryo state for more details.
- 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 Brain UI later at *Data Sources > Network > Sensors > Sensor Configuration > CLI Password (Sensors)* if you wish to keep them in sync.
    - You will be asked to change the password after initial login to paired and updated vSensors.

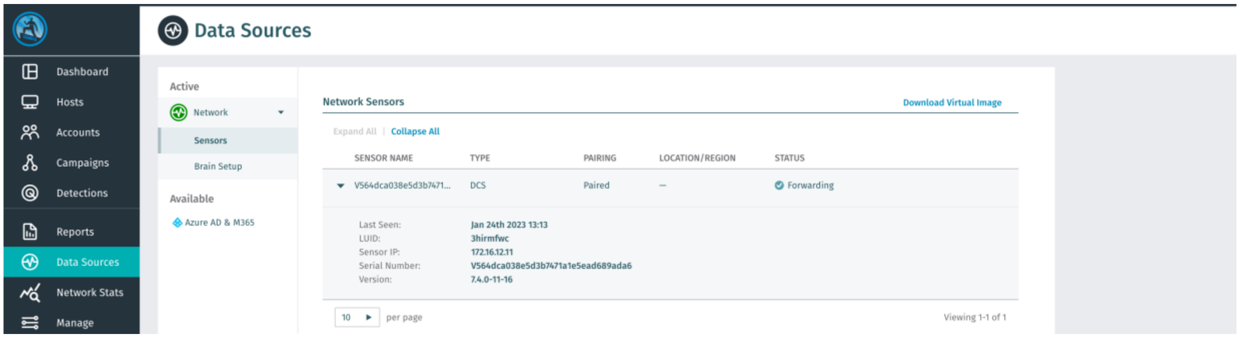
Example:

```
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 VMware vSensors

- vSensors do not offer a web UI.
  - The VectraPlatform (Brain) has the GUI for vSensor management at *Data Sources* > *Network* > *Sensors*.
  - Some configuration of the vSensor can be done at the CLI of the vSensor using the "**vectra**" user.
- VMware 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.



Data Sources

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

- **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.
- 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

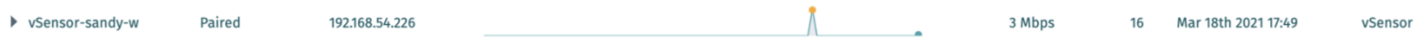
- The vSensor image downloaded from the Brain will use, by default, the Brain's IP address for pairing.
- You will need to set the *Data Sources > Network > Sensors > Sensor Configuration > Sensor Pairing and Registration* **"Pair using DNS name"** option to generate the virtual machine image that points 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:

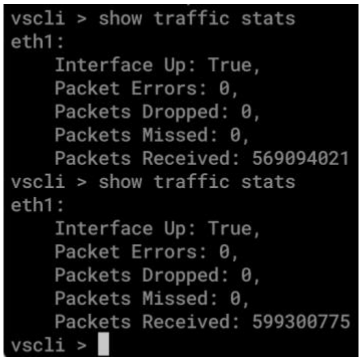
- Vectra Platform Network Traffic Recommendations

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.
  - Please note that this command will only function after the vSensor has been paired and updated from the Brain. For details, please see the earlier guidance about the embryo state of vSensors immediately after initial deployment.
- 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