# CX10K Hands on Lab User Guide

by Toby Makepeace / Don Zaine

## Abstract

This is the user workbook for the DPU test-drive hands on lab

Makepeace, Toby
Toby.makepeace@amd.com

# Contents



Makepeace, Toby
Toby.makepeace@amd.com

# Doc version

| | | | |
|---|---|---|---|
| 0.1 | Toby Makepeace | | 3/3/2025 |
| 0.2 | Don Zaino | | 4/02/2025 |
| … | Don Zaino | | 4/07/2025 |
| 0.9 | Toby Makepeace / Mike McSpedon | Update and combine multiple edit versions | 5/2/2025 |

## Key Abbreviations

These will help you reference any abbreviations you encounter while reading through the lab Guide

**DPU** = Data Processing Unit

**vDS** = VMWare Distributed Switch

**DSM** = Distributed Services Module

**DSS** = Distributed Services Switch (the CX 10000)

**ELK** = Elasticsearch Logstash Kibana

**PSM** = Policy and Services Manager

**VM** = Virtual Machine

**WS** = Workstation

# Learning Objectives

Upon completion of this Hand's on Lab you will be able to:

1. Understand the value and benefit of a DPU enabled smart switch
2. Interpret VMware networking, PSM policy and ELK visibility
3. Familiarize yourself with the layout of PSM and ELK interfaces
4. Understand how micro- and macro-segmentation can be achieved on east/west workloads on the CX 10000 which can help you achieve your Zero Trust architecture goals in your datacenter architecture
5. Have a working knowledge of basic base policy features of PSM

# Getting Started

This guide provides a high-level overview of the capabilities of the CX 10000 Distributed Services Switch (DSS). Its purpose of this guide is to provide guidance and inspiration for experimenting within your own controlled, self-contained lab environment either through the PSM or via the API. Since each hosted environment of the DPU test-drive will vary, the course leader will explain any local variations or changes. This course aims to familiarize you with the features that can be configured and managed via PSM. Please **explore**, break, learn and enhance your knowledge of the products during this time.

# Scenario

Throughout this lab activity, you will learn how the HPE Aruba CX1000 with AMD Pensando DPU solution can enable a next gen data center architecture.

By inserting stateful service capabilities in a data center fabric, security and visibility are now integrated into your environment and placed closer to where applications and workloads live without any impact on existing network architecture or software configuration changes.

In this lab you will learn how to review and understand VMware networking, set up micro-segmentation and firewalling services policy through the Pensando Policy and Services Manager (PSM) and gain visibility to your data in the Elastic - ELK stack.

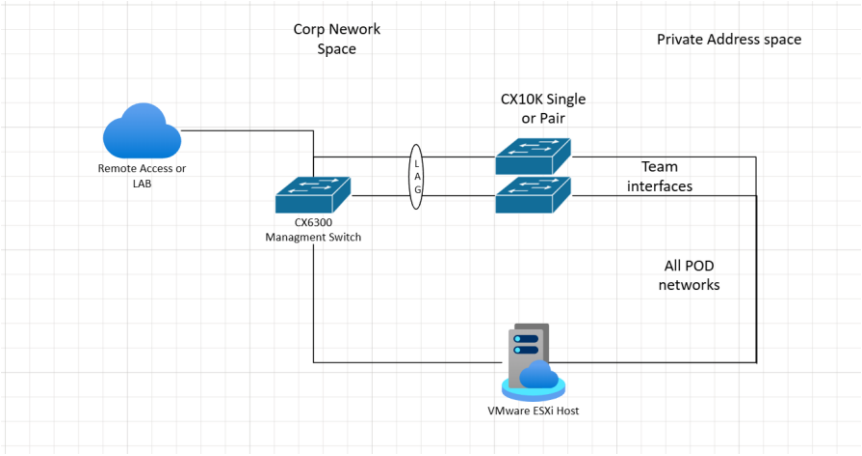
The CX10k includes a pair of Pensando/AMD DPU's. While these DPUs support multiple programmable functions like stateful firewalling, encryption, NAT, and real-time telemetry, this lab focuses on firewalling, security and policy enforcement.

**PSM** = Pensando Servies manager is a programmable, secure and highly available centralized system for managing infrastructure policy. It is designed to establish and manage policies for every CX 10000 Distributed Services Switch (DSS).

**ELK** = Is a suite of open-source tools for log management, analysis, and visualization comprised of Elasticsearch, Kibana, Beats, and Logstash (also known as the ELK Stack). It allows you to take data from the CX 10000 switch in any format, then search, analyze, and visualize that data.

# Topology

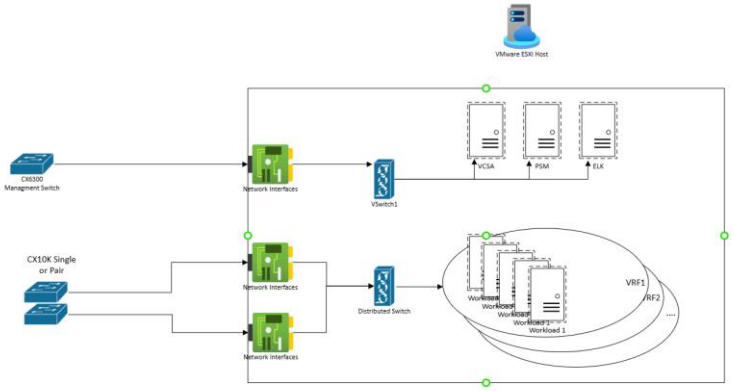
## Understanding the Lab Environment




The above diagram shows an overview of the lab topology; the actual hardware might change, but it is shown as an example.

The lab is designed so that all pod access is controlled, and you have a safe environment to work and learn. Access to the environment can be provided by either remote access or classroom access (when onsite). Everything you access will be hosted on the VMware ESXi host. Your access to the services on the ESXi host is via the management switch.

The VMware ESXi has 3 network interfaces configured. One – north to the management switch where you access is granted, and then Two – southbound to the CX10k that are used for the services testing.




The applications you have access to as part of the LAB are the vCenter server (VCSA), PSM, and ELK. Via the vCenter server, you can access your pod and configure your workloads as required.

The pod has been set up with a default configuration, you will be required to access the workloads and configure them as you require to carry out the tests. It is expected that you will use applications like PING/SSH/CURL/IPERF to test the features.

The lab will take you through things like the moving of the workloads to a PVLAN Isolation network to perform “micro segmentation”, and/or moving of the workloads to a different VLAN to perform “macro segmentation”.

The lab guide has been written as an example and to explain what is available, but it is really to grant you free access to test the scenarios that are important to you. Please feel free to expand the lab test cases to meet your own requirements.

## Username

The username you have been assigned is based on your pod ID. This allows for segmentation of your environment from other students. However, as this is running on shared infrastructure for the vSphere, PSM and ELK environments we ask that everyone stays within their own pod.

To access any items specific to your pod simply replace the <POD> in the examples with your given assigned pod ID.

Example:

| POD | PSM user | VCenter user | ELK User |
|---|---|---|---|
| 1 | vrf1_user | vrf1_user@vsphere.local | vrf1_user |
| 2 | vrf2_user | vrf2_user@vsphere.local | vrf2_user |
| …. | | | |

### PSM Username

There are two accounts below that allow you to login to PSM. The read-only account grants access to the entirety of PSM, while the second account has restricted write permissions, allowing you to modify your network and policies within your pod.

The read-only account is optional for exploration of all that PSM has to offer , which can be useful tobetter understandf the layout and options available along with how the Role Based Access works within PSM.

| Username | Password | Role |
|---|---|---|
| read-only | Pensando0$ | Read-only |

| | | |
|---|---|---|
| vrf<pod>_user | Pensando0$ | vrf(pod) |

## PSM Role Read-only.

The read-only role has full read permissions to all object's inside of PSM, it does not have any write, update or delete permissions.

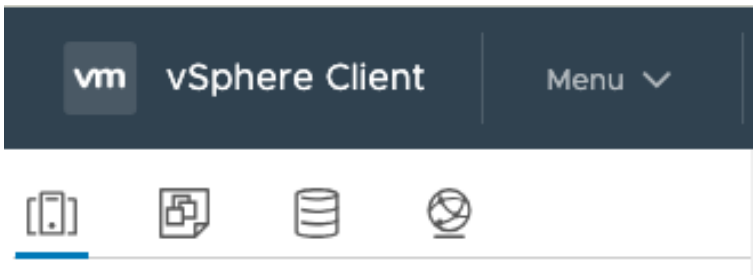
## PSM Role vrf (pod)

The (pod) role has full read access to most of the object tree. However, for VRF, Networks, and Security policies, it only has read-write permissions for the assigned objects.

You will primarily use the vrf<pod>_user account throughout the lab. However, in some cases you may want to log in with the full read-only account. When doing so, it is best to use and incognito window.

# VMware

For those unfamiliar with the vSphere client, the symbols below represent the main areas you will be working in. From left to right they are:





Hosts and Clusters


Virtual Machines and Templates


Storage

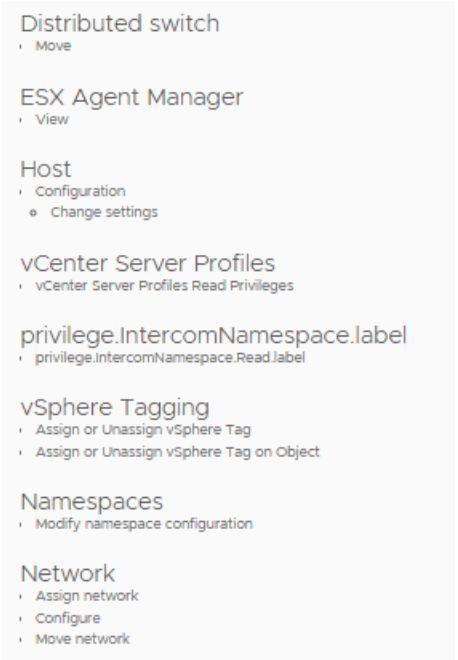
Networking

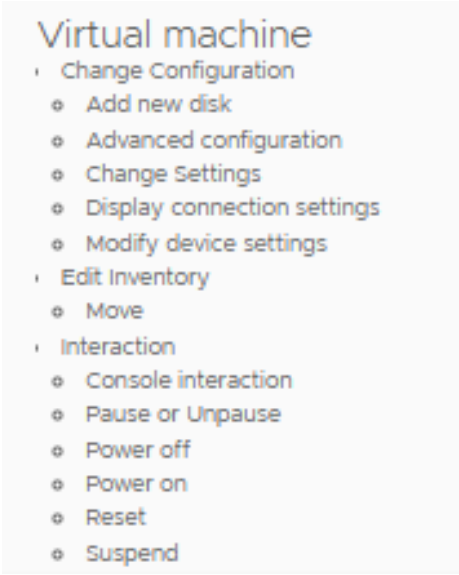
You will also receive credentials to access a shared vCenter server environment. This account will have permissions scoped to the Port groups, Resource Pools and virtual machine workloads assigned to you.

| Username | Password | Role |
|---|---|---|
| vrf<pod>_user@vsphere.local | Pensando0$ | TestDrive |
| readonly@vsphere.local | Pensando0$ | Read only |

## Role TestDrive.

The “TestDrive” role includes the following assigned permissions:




VMware Role read-only

The read-only role has full read permissions to all objects inside vSphere, it does not have any write, update or delete permissions.

# Workstations (VM's)

## VM Accounts

Each pod is provisioned with three workstations. These VM's are initially assigned IP addresses from the primary network range and vlan but may be moved or migrated across different networks to test and validate various policy changes and configurations.

| User | Password |
|---|---|
| tc | VMware1! |
| root | VMware1! |

## Commands to edit a VM's IP

To change the IP address of the workstation, the following command options are available:

### *Option* 1.

```
sudo ifconfig eth0 <IP> netmask <mask> up
sudo route add default gw <gateway>

example.

sudo ifconfig eth0 192.168.11.11 netmask 255.255.255.0 up
sudo route add default gw 192.168.11.1
```

### *Option* 2.

```
sudo /opt/set-ipv4-address.sh <IP> <mask> <broadcast> <Gateway> <DNS> <domain>


example.

sudo /opt/set-ipv4-address.sh 192.168.11.11 255.255.255.0 192.168.11.255
192.168.11.1 192.168.101.1 testdrive
```

## VM Networks and subnets

Each pod has been allocated three VLANs and three subnet ranges specific to the pod.

The model for the networks and subnets is based on the pod number.

VRF – vrf<pod>

| Network | Subnet | Mask | Gateway | DNS | Domain |
|---|---|---|---|---|---|
| VLAN<pod>1 | 192.168.<pod>1.0 | /24 | 192.168.<pod>1.1 | 8.8.8.8 | Testdrive |
| VLAN<pod>2 | 192.168.<pod>2.0 | /24 | 192.168.<pod>2.1 | 8.8.8.8 | Testdrive |
| VLAN<pod>3 | 192.168.<pod>3.0 | /24 | 192.168.<pod>3.1 | 8.8.8.8 | Testdrive |

For example, Pod 1.

| VRF – vrf1 | | | | | |
|---|---|---|---|---|---|
| Network | Subnet | Mask | Gateway | DNS | Domain |
| VLAN11 | 192.168.11.0 | 255.255.255.0 | 192.168.11.1 | 8.8.8.8 | Testdrive |
| VLAN12 | 192.168.12.0 | 255.255.255.0 | 192.168.12.1 | 8.8.8.8 | Testdrive |
| VLAN13 | 192.168.13.0 | 255.255.255.0 | 192.168.13.1 | 8.8.8.8 | Testdrive |

## Workstation Addresses

Each pod has three workstations deployed on the first network and the following Port-Group

| VRF – vrf<pod> | | | |
|---|---|---|---|
| Workstation | IP Address | Port Group | Example |
| Workstation<pod>1_11 | 192.168.<pod>1.11 | <pod>1_10<pod>1_PRO | Workstation11_11 / 192.168.11.11 / 11_1011_PRO |
| Workstation<pod>1_12 | 192.168.<pod>1.12 | <pod>1_10<pod>1_PRO | Workstation11_12 / 192.168.11.12 / 11_1011_PRO |
| Workstation<pod>1_13 | 192.168.<pod>1.13 | <pod>1_10<pod>1_PRO | Workstation11_13 / 192.168.11.13 / 11_1011_PRO |

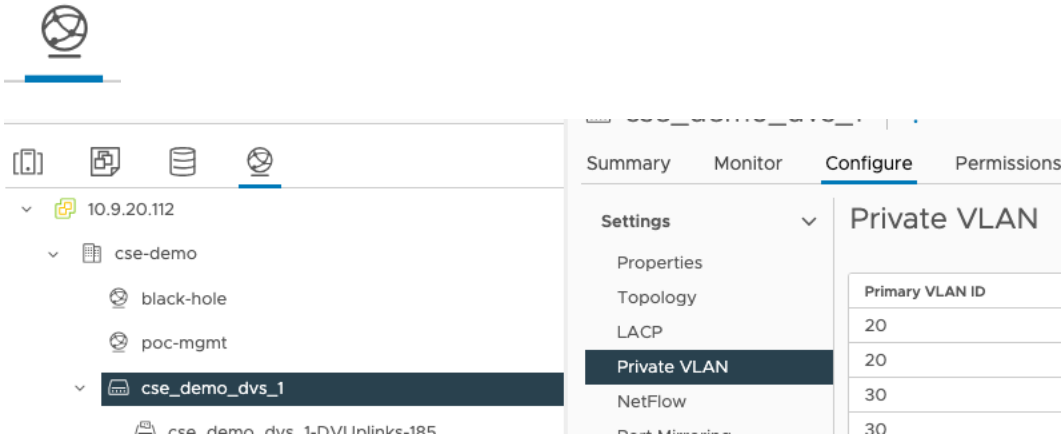
# VMWare Networking Exploration

The VMware Distributed Switch and Port groups have been pre-configured, each individual pod space has been allocated a total of 6 Port groups with 2 port groups per vlan based on a standard PVLAN configuration. In the micro-segmentation scenario, we use PVLAN isolation to get the traffic from the Distributed switch port group out and up to the CX 10000, allowing traffic to be visualised and managed.

The “Distributed switch” created is called “VRF-DEMO”, and the configuration can and should be reviewed with the read-only account to gain a better understanding of the environment.  (use the readonly@vsphere.local account to review)

## vSphere Distributed Switch

In the **Networking** tab under your provisioned DVS in the configure section of the distributed switch you can find the Private VLAN section. Each primary Network VLAN ID will require a secondary VLAN configured under it and selected for Isolated mode.




In this setup, Isolated mode is used to direct traffic to the CX10k’s SVI interfaces for management. While community mode is an available option for PVLAN configuration it is not used in this scenario.

## vSphere Port-Groups

The configuration of the Isolated and Primary networks must align with the network settings on the CX10k’s.

| Primary VLAN | Secondary VLAN | Primary Port Group | Isolation Port Group |
|---|---|---|---|
| <pod>1 | 10<pod>1 | <pod>1_10<pod>1_PRO | <pod>1_10<pod>1_ISO |

| | | | |
|---|---|---|---|
| <pod>2 | 10<pod>2 | <pod>2_10<pod>2_PRO | <pod>2_10<pod>2_ISO |
| <pod>3 | 10<pod>3 | <pod>3_10<pod>3_PRO | <pod>3_10<pod>3_ISO |

*PRO designates a promiscuous vlan type

*ISO designates an isolated vlan type

Example

| Primary VLAN | Secondary VLAN | Primary Port Group | Isolation Port Group |
|---|---|---|---|
| 11 | 1011 | 11_1011_PRO | 11_1011_ISO |
| 12 | 1012 | 12_1012_PRO | 12_1012_ISO |
| 13 | 1013 | 13_1013_PRO | 13_1013_ISO |

### PVLAN Modes

| | |
|---|---|
| Promiscuous | A port that is a member of the primary VLAN. These ports can send packets to all ports of the primary VLAN and ports of associated isolated and community VLANs. Promiscuous ports are used to communicate outside the PVLAN domain. |
| Isolated | A secured VLAN where hosts cannot communicate with each other through L2 switching. |
| Community | VLAN where hosts can communicate with each other through L2 switching. There can be multiple community VLANs associated with a primary VLAN. CX 10000 does not support this PVLAN mode. |

# PSM

The majority of your tasks will be performed through PSM, the management platform responsible for the security components of the CX 10000.

The PSM can be deployed as an OVA image on VMware, VHD on Hyper-V, or qcow image on any KVM based platform, as previously mentioned, it has already been deployed for you in this lab environment, and the two DSSes (CX 10000s) have already been admitted to PSM.  You can see this in PSM explore the PSM UI and find where the switches are healthy and admitted. Hint: DSS.

Additional information and Options regarding PSM can be found in the PSM user-guide at https://networkingsupport.hpe.com/.

## PSM VRF and Network

To match the VLANs configured on the CX 10000 switch and the VMware environment, every pod is allocated 3 networks in PSM within the dedicated pod VRF you have been assigned.

The networks within PSM are linked to the  CX 10000based on the VLAN ID, VRF ID and must match the primary (Promiscuous VLAN) in the PVLAN configuration on the VMware environment.

| VRF – vrf<pod> |
|---|

| Network | VLAN ID | VRF | Ingress Policy | Egress Policy |
|---|---|---|---|---|
| VLAN<pod>1 | <pod>1 | vrf<pod> | | |
| VLAN<pod>2 | <pod>2 | vrf<pod> | | |
| VLAN<pod>3 | <pod>3 | vrf<pod> | | |

Example from Pod 1.

| VRF – vrf1 |
|---|

| Network | VLAN ID | VRF | Ingress Policy | Egress Policy |
|---|---|---|---|---|
| VLAN<pod>1 | <pod>1 | vrf<pod> | | |
| VLAN<pod>2 | <pod>2 | vrf<pod> | | |
| VLAN<pod>3 | <pod>3 | vrf<pod> | | |

## PSM Security Policies

There have been 6 pre-defined security policies created for each pod to begin your work with, they can be assigned to either Egress or Ingress on the VRF or Network.

The names allocated to the policies as part of the lab are available to use in any way you see fit. It is recommended that you use the first policy as the active policy and your newly created policy as the policy you will migrate your environment to.

| Policies | | Example Pod 1 |
|---|---|---|

| | | | |
|---|---|---|---|
| vrf<pod>_vlan<pod>1_First | vrf<pod>_vlan<pod>1_new | vrf1_vlan11_First | vrf1_vlan11_new |
| vrf<pod>_vlan<pod>2_First | vrf<pod>_vlan<pod>2_new | vrf1_vlan12_First | vrf1_vlan12_new |
| vrf<pod>_vlan<pod>3_First | vrf<pod>_vlan<pod>3_new | vrf1_vlan13_First | vrf1_vlan13_new |

⚠️ Additional information regarding Policy enforcement from a PSM/DPU perspective regarding ingress and egress definitions can be found here Pg.94
https://arubanetworking.hpe.com/techdocs/Pensando/AMD_Pensando_PSM_for_DSS_Guide-1.100.2-T.pdf

# Activities

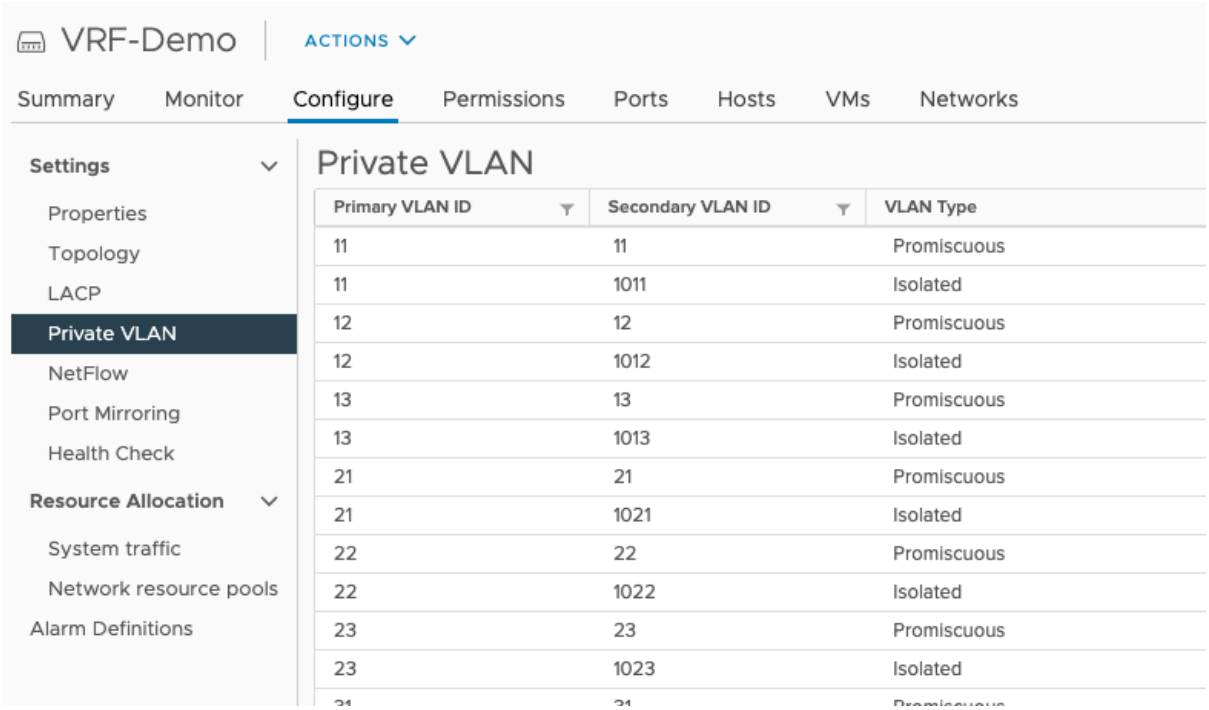
**The URL for the Vcenter server will be provided by the trainer, as it is unique to every lab.**

## Exploring VMware networking

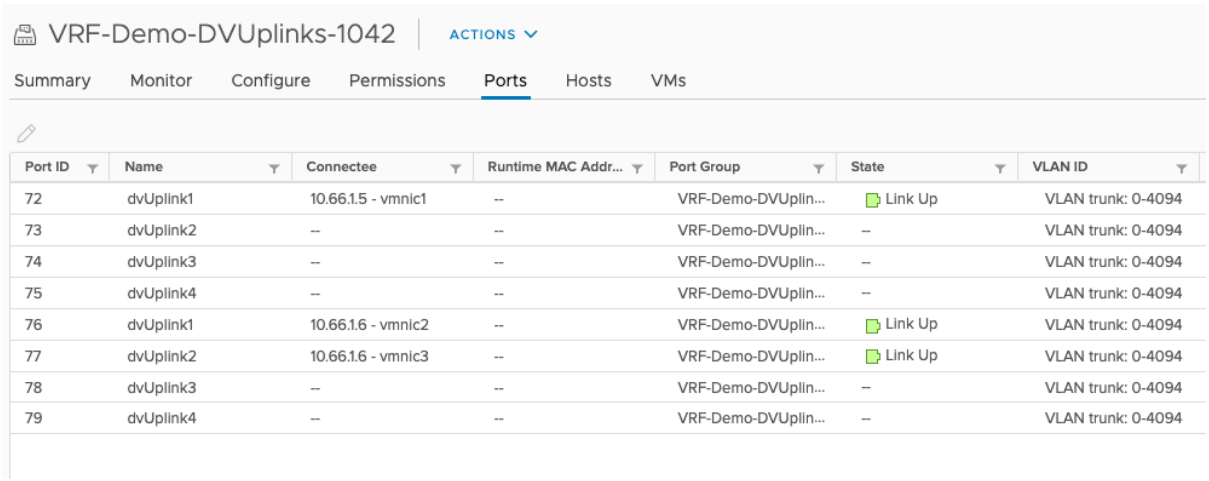
Review The VMware configuration using the “read-only” account, to get a better understanding of the layout.

| Username | Password | Role |
|---|---|---|
| readonly@vsphere.local | Pensando0$ | Read only |

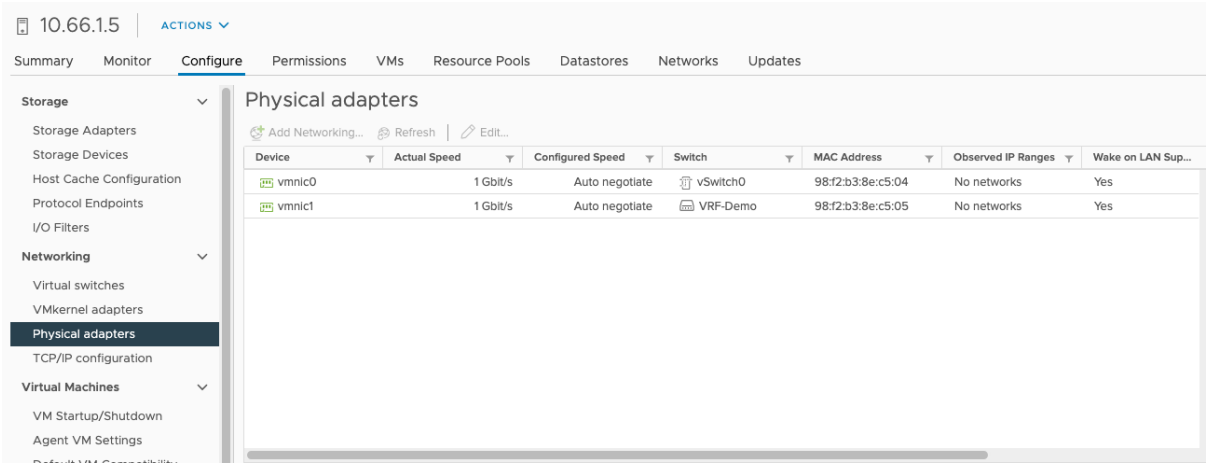
1. Go to the **network tab** and select the “VRF-Demo” Distributed switch.
    a. Look at the configure – Private VLAN section, you will see the VLAN mappings.




b. Look at the VRF-Demo-DVUplinks




c. Check the ports allocated and then go to the host and look at the physical host interfaces and LLDP config.




What do you see?

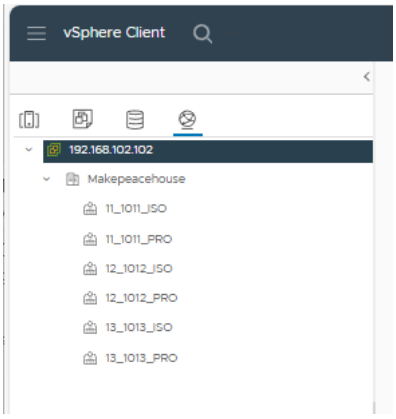
Your vSphere Networking/Physical adapters section should look similar to the above, ask questions if you do not understand or need clarification on anything.

**This is the same for other Hypervisors and other physical switches.**

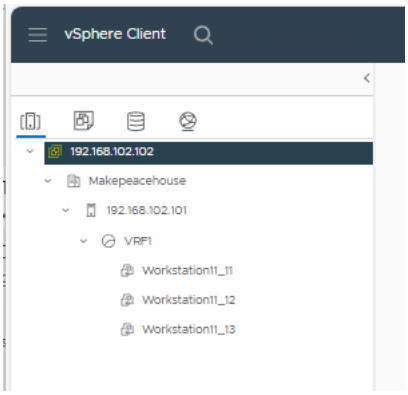
## Access vSphere workloads

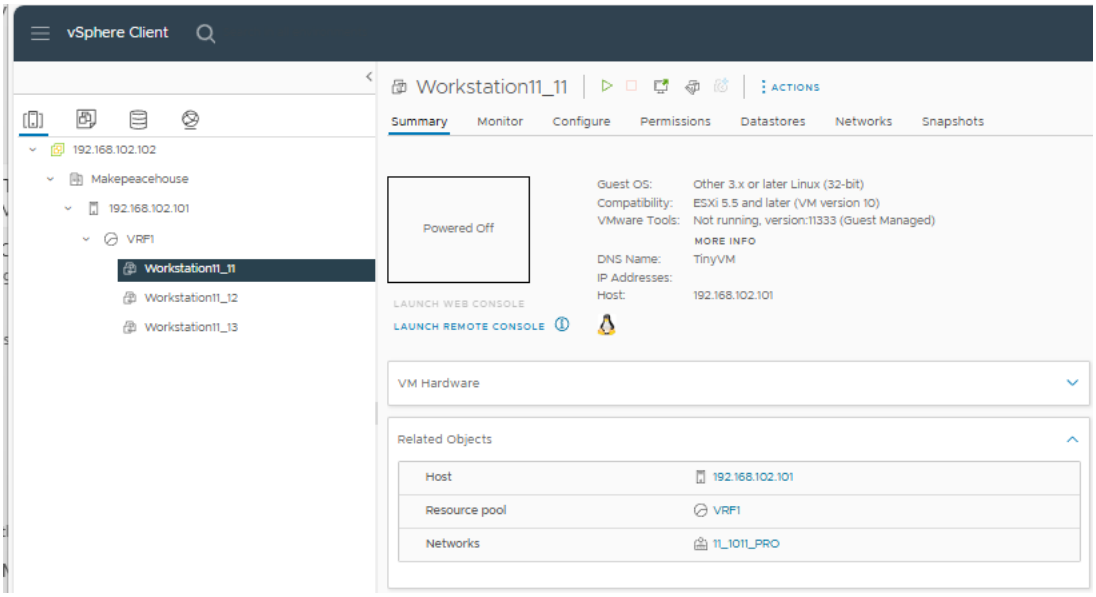
The vm console can be accessed via the Vcenter server by selecting your desired vm and launching the Web Console.

1. Login into Vcenter with you vrf<pod>@vsphere. local account. Review the **VMware networking** you have been allocated.



2. Look at the **workloads** you have been allocated and the networks they have been assigned.




3.

In a later step you will be assigning your workloads to different Port-groups, but to start with we recommend leaving them on the default Port Group's they are bound too.

## Confirm VM Base Networking

If you happened to edit any of the original Port Group configuration you would also need to edit the IP address of the workloads so they can access the corresponding network. At this point, if no changes have been made to port groups or IP addressing then the workloads should be able to communicate.


1. You can perform simple ping tests to validate this.

Refer to the commands in the section Commands to change IP and test that you can access the SVI interface on the CX10k switch and the any other workstations.

### Testing VM Networking

To validate network connectivity, ensure you can successfully ping both your pods VLAN gateway and the IP addresses of the peer VM's. if connectivity fails, troubleshoot and resolve the network issue before proceeding to the next step.

At this stage, no security policies are enforced, and any traffic restrictions would be most likely due to network or vSphere misconfiguration

## Testing / Exploring in PSM

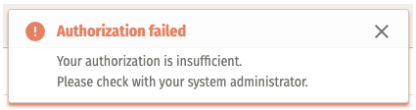
**The URL for the PSM will be provided by the trainer, as it is unique to every lab.**

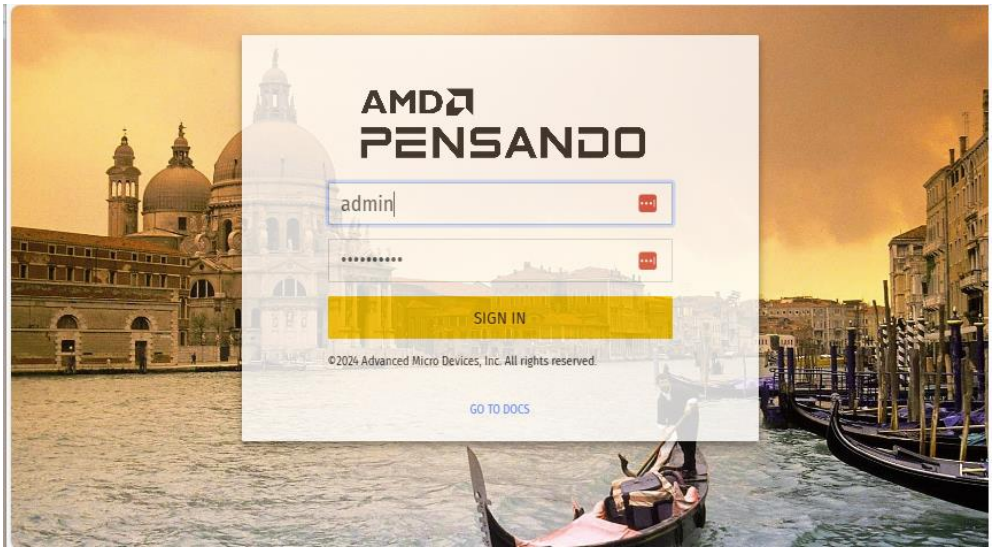
### Enable Policy

To begin enabling policy enforcement, we ned to use the assigned axis pod cross launch link or IP for PSM and place it in a browser to continue to login with you vrf<pod> account.

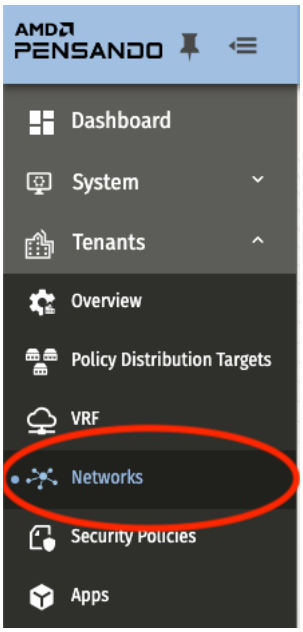
| Username | Password | Role |
|---|---|---|
| vrf<pod>_user | Pensando0$ | vrf(pod) |

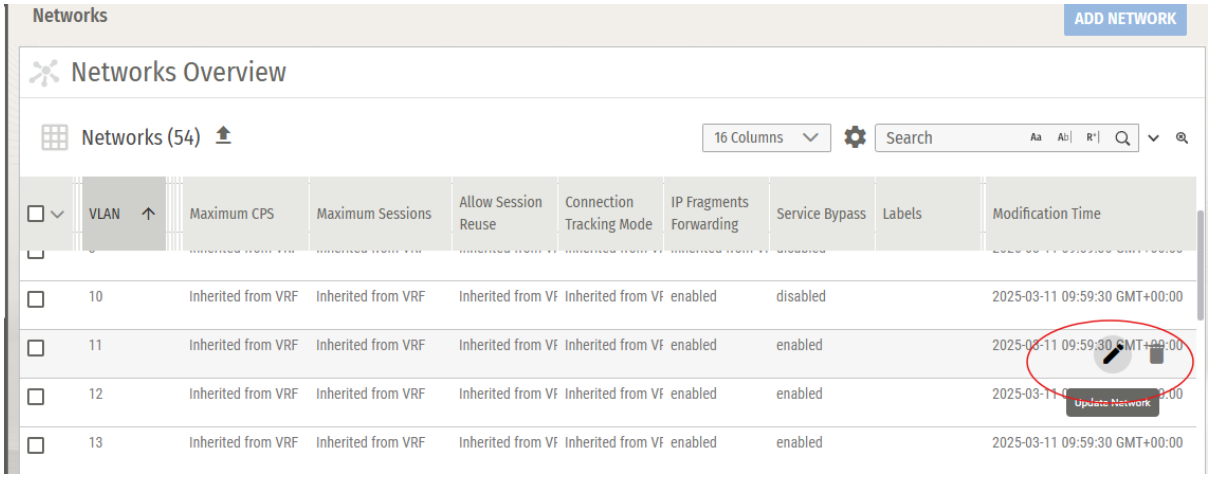
⚠️ Note: You may see an ‘Authorization failed’ notification in the top-right window corner when working in the PSM GUI. This can be ignored, as it is related to the assigned permissions.




Policy creation and enforcement begins under the Tenant portion of the menu in PSM, specifically under the ***Network*** section. For the initial phase we focus on the network section.

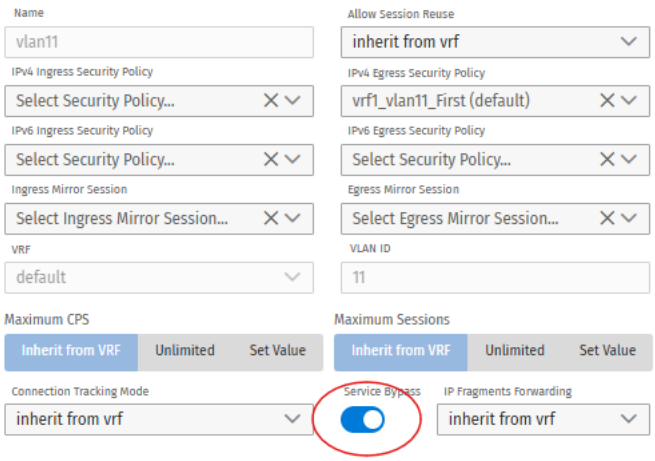





1. Select the pencil option on the right end of the line to edit any network configuration.

From the **network** overview section, you can see the policies assigned on ingress or egress, and the service bypass mode is “enabled”.

(All default policies have been created with only an allow rule to get started).




2. Disable “service bypass”, and assign a policy, and save.

> ⚠ **Note:** Enable/Disable service-bypass. When enabled, the distributed services will be disabled. Blue means enabled – meaning no polices will be enforced

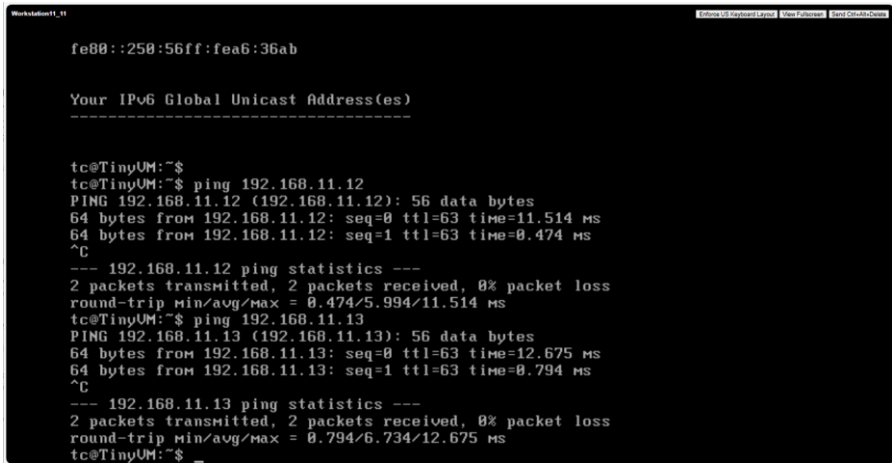
## Data Visibility in PSM

Once a policy is enabled, all traffic traversing the switch is subject to the policy enforcement. Because logging has already been configuired, all corresponding logs will

be automatically forwarded to both the PSM and the pre-configured ELK stack for analysis and monitoring.

You can now generate test data between the workloads and out to the internet (if accessible).

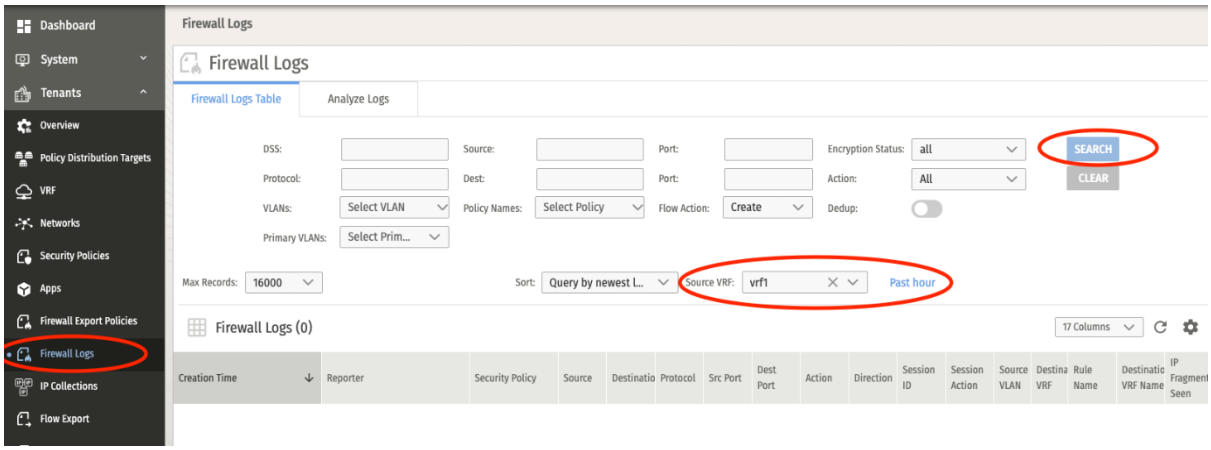
**Ping works fine to start**




## Viewing logs in PSM

Moving through the left-hand menu in the PSM GUI, go to the **Firewall Logs** section.

1. Select Firewall Logs from the menu.
2. Select the VRF you are interested in in the drop-down menu.
3. Next to the selected VRF, choose the time period you are interested in and apply it to the search.



**Do you see any log data?**

After disabling service bypass the Logs get forwarded to the PSM instance. This happens either every 5 minutes or 10,000 log events whichever happens first so, there may be a slight delay.

**What logs do you see and not see? Why?**

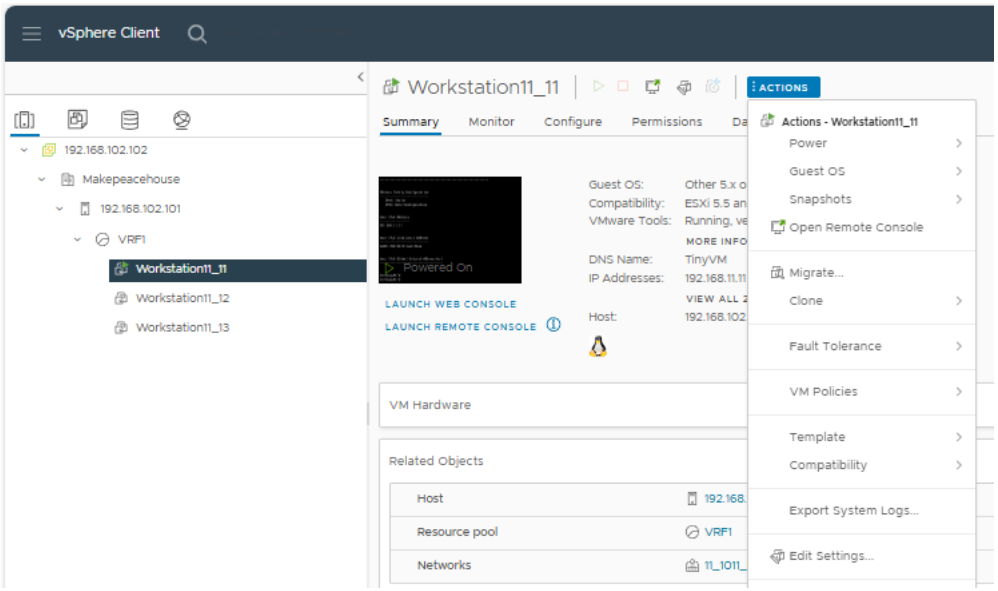
- Talk to the group and the trainer to understand what you see and do not see and why.

## Configure micro-segmentation

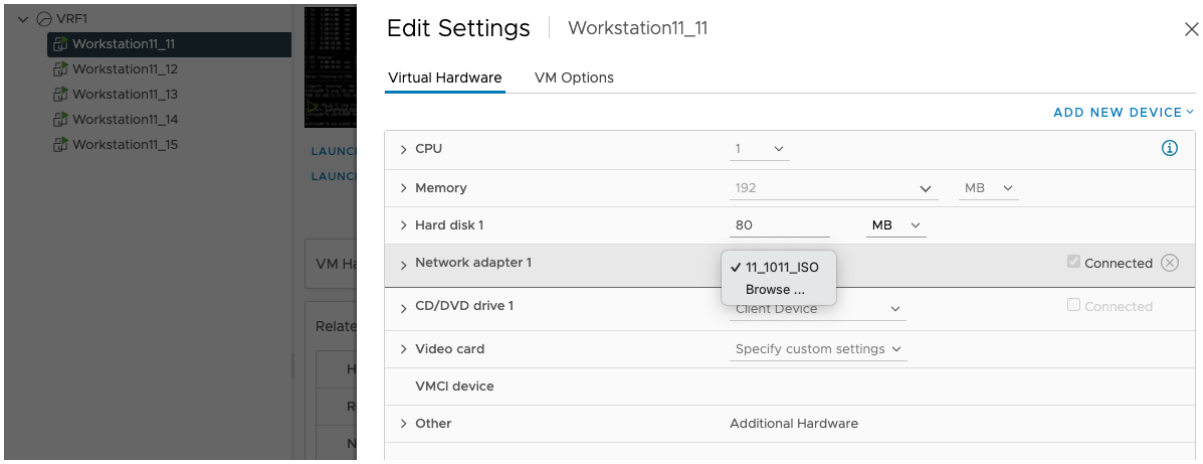
Currently, the attached Port-Group these Workloads have been assigned to is on a promiscuous network, meaning that switching is performed locally within the virtual switch inside the hypervisor and therefore this traffic never needs to leave the ESXi host.

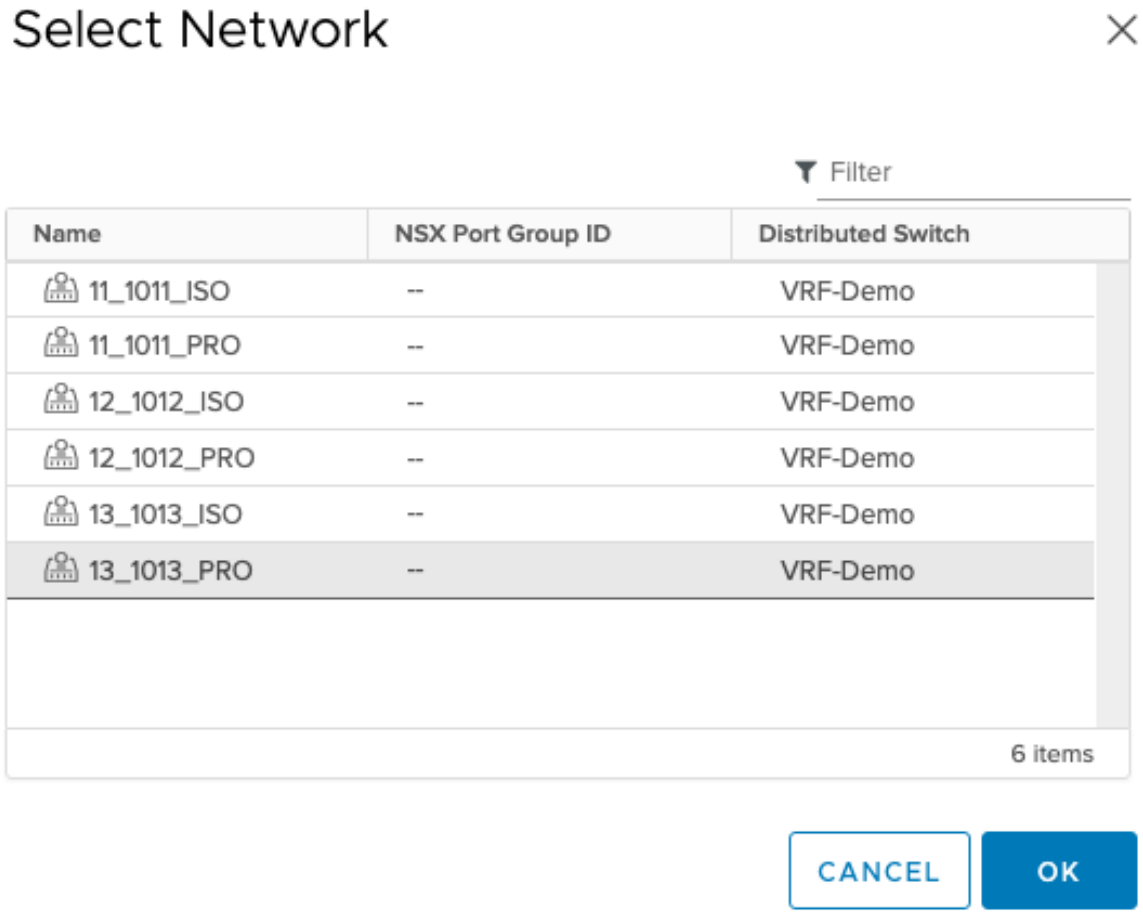
To enable visibility and management of the traffic, the workload must be configured to communicate over a PVLAN in isolated mode. This ensures that traffic is forwarded upstream out of the host to the switch for inspection.

1. In vSphere Edit the selected **workload** settings.



2. Relocate the selected workload vm to your isolated VLAN by browsing the available networks and selecting the isolated version of that network that corresponds to your host's original network.




3. Repeat this for any or all your workload vm's and continue to View_in_PSM

## View Visibility data in ELK

**The URL for the ELK stack will be provided by the trainer, as it is unique to every lab.**

Logs from the CX10K can be forwarded to any external syslog collector in real time.

The ELK stack has been selected as the demonstration tool for this lab due to its availability and popularity as a community edition and free open-source suite.

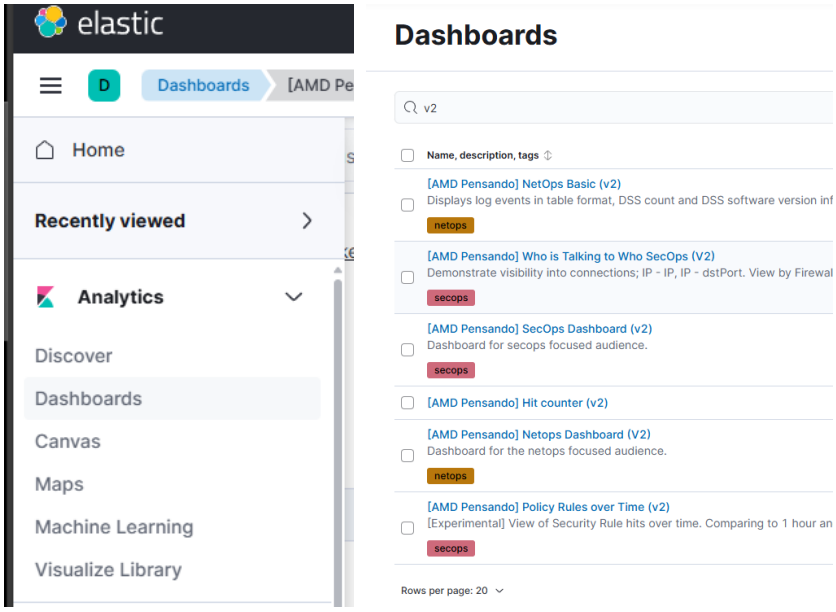
There are many paid and licensed tiers for ELK along with multiple other vendors who provide syslog collectors that can be used in conjunction with the CX10K. You are not tied to using ELK it is just what we have found to fit best for our lab purposes.

*The ELK stack like the PSM and Vcenter instance is shared for all pods.

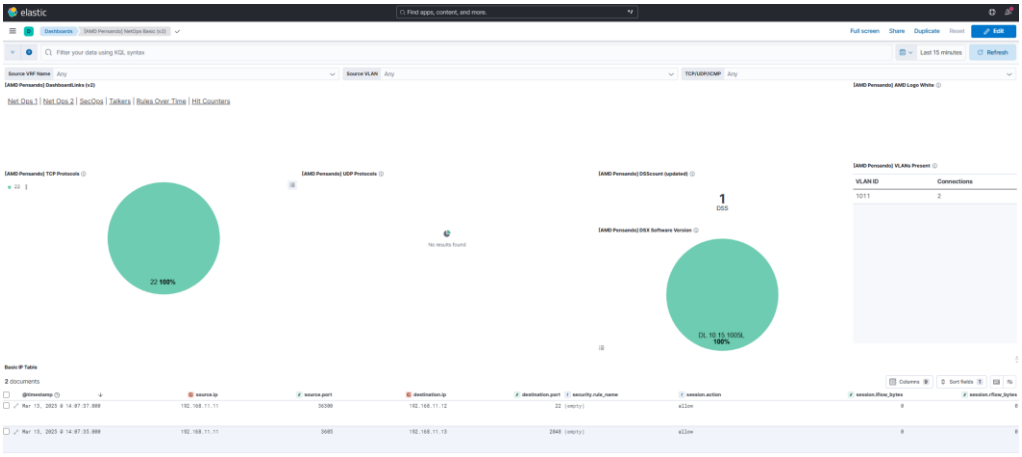
| Username | Password | Role |
|---|---|---|
| vrf<pod>_user | Pensando0$ | |

1. To access the ELK server, go to the url:
   http://<ELK IP>:5601/
   (credential: vrf<pod>_user - Pensando0$)

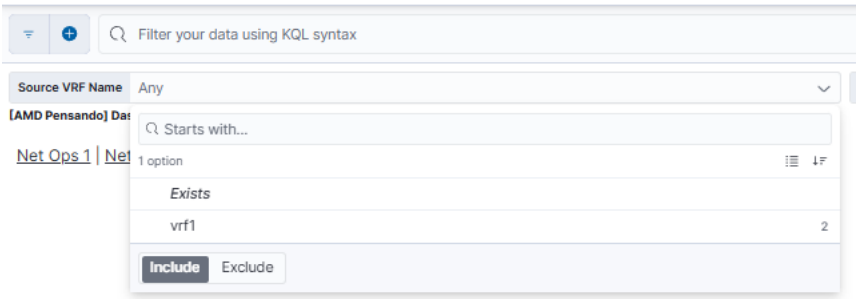
2. Select the **Dashboard** option, and search for AMD.



3. Select the “[AMD Pensando] NetOps Basic v2” **Dashboard**.



4. Can you see your pod logs?  Can you filter the logs to your pods VRF.




5. What logs do you see and not see? Why?
6. Explore the ELK dashboard by viewing your data through the various dashboards and visualizations. Continue to revisit the dashboards while performing other testing to monitor any changes and gain insights into how your traffic is behaving and what your traffic is doing.

## PSM Base Features Testing

There are several base features that exist in PSM regarding firewalling that must be understood to help you effectively apply and construct policies. The tasks we will focus on are:

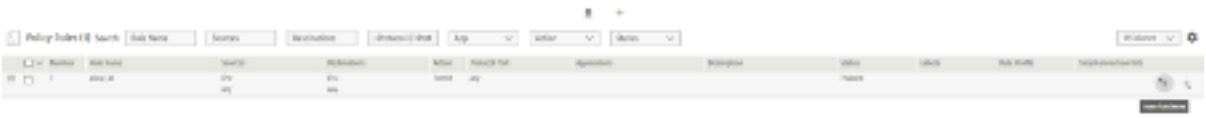
- Block ICMP & Service Bypass
- IP-Fragmentation support
- Connection Tracking
- Session Reuse
- Block SSH
- Block traffic based on IP-Collections
- Block traffic based on VMware tags
- Block IPERF

### Task 1 - Block ICMP between two workloads

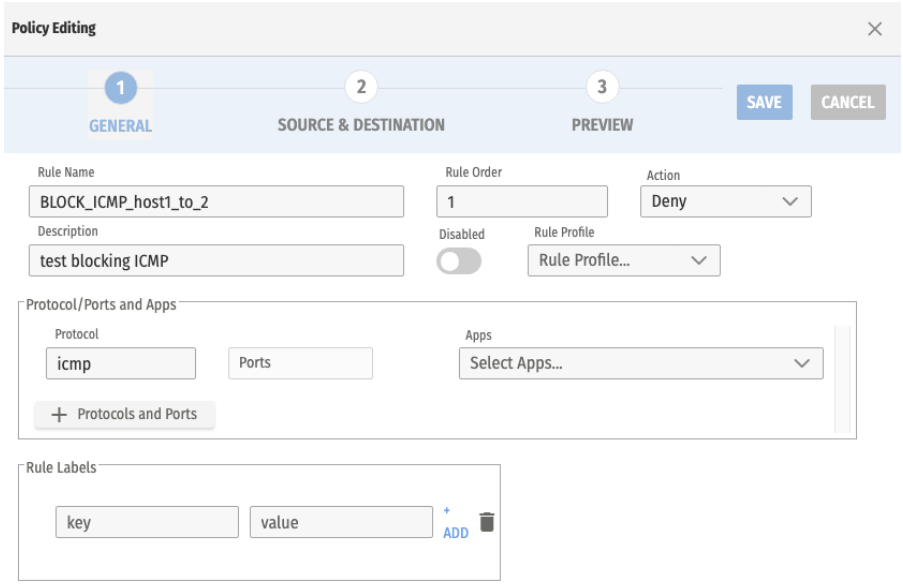
Make sure you have walked through the previous steps before proceeding.

You will now configure a simple deny ICMP policy between your workload1 and workload2 (directional) in your POD as a test of the next few features.

In the **Security policies** section find your security policy <vrfx_vlanxx_first>, select the pencil on the right side to “Edit Policy”. Under policy Rules click the “+” to add a rule to your policy. You may select either the “+” on at the top of the rules or the “+” at the end of the line, for the insertion of a rule above or below the existing “allow_all rule”.

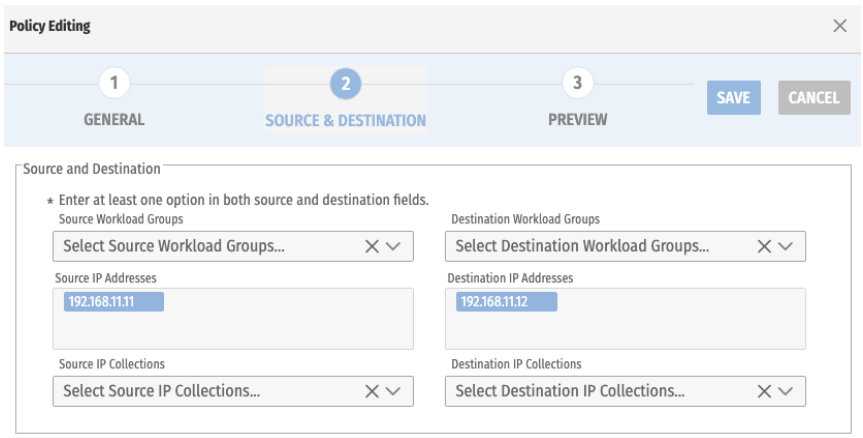


Step 1- You need to provide a “Name” for the rule, verify the rule order number is 1, change the action to “Deny”, add a description if needed and type icmp in the protocol box. Enable rule is default



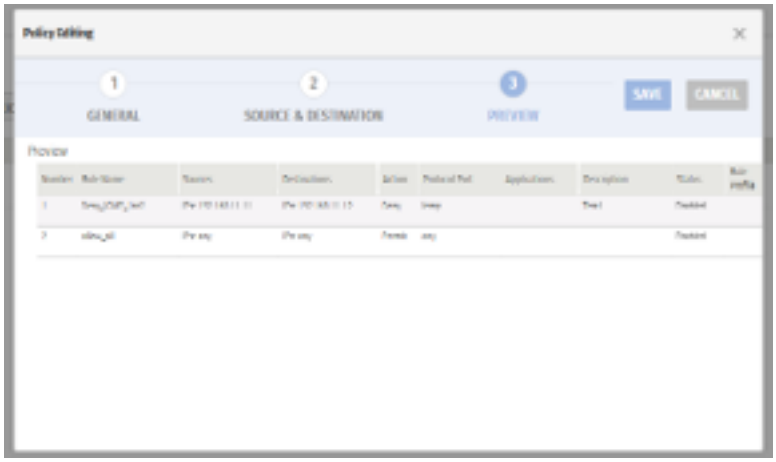

There are also additional options around the use of port, protocol or Pre-defined Apps that may be used. (Apps are L4 based on port and protocol only).

Step 2 - In the policy editing window define the source IP and destination IP of your chosen workloads. You have the option to use only the IP or a CIDR range from 0 – 32 if you wish.




Step 3 - Preview the rule to make sure it meets the requirement and click save and Save policy at top of screen.

- Proceed to test if you can ping from host 1-2 and 2-1. Also review the logs in PSM and ELK
- Do they match what you expect?



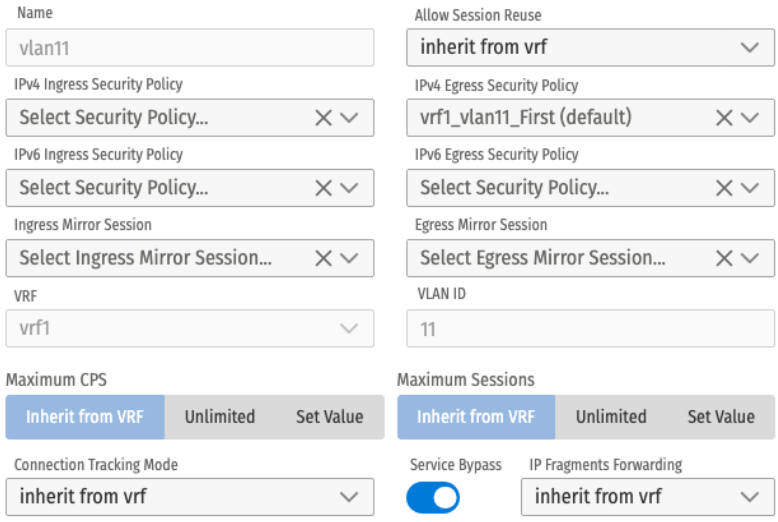

Learning Check Task 1

1. Navigate and Overview of the PSM Gui
2. Explore Security Policies
3. Created a security policy and all its attributes
4. Verified a working policy rule
5. Verified log collection on PSM and ELK

## Task 2 - PSM service Bypass

Service bypass is a feature that can be enabled to allow bypass of a VLAN from policy management.

Continuing from the Block-ICMP rule we just created in the previous section. From the **Networks section** go to your vlan and enable the service bypass slider by toggling it to ON (it will turn blue), click save and ok on pop up and then re-run the ping tests and check/view the logs.




⚠ Note: Enable/Disable service-bypass. When enabled, the distributed services will be disabled.

- Is the output as expected?
  - Are you seeing Logs
  - Is the Enforcement correct

Learning Check Task 2

1. Using the PSM GUI to verify a policy
2. Working with service bypass feature
3. Verifying log visibility

## Task 3 - PSM Fragmentation

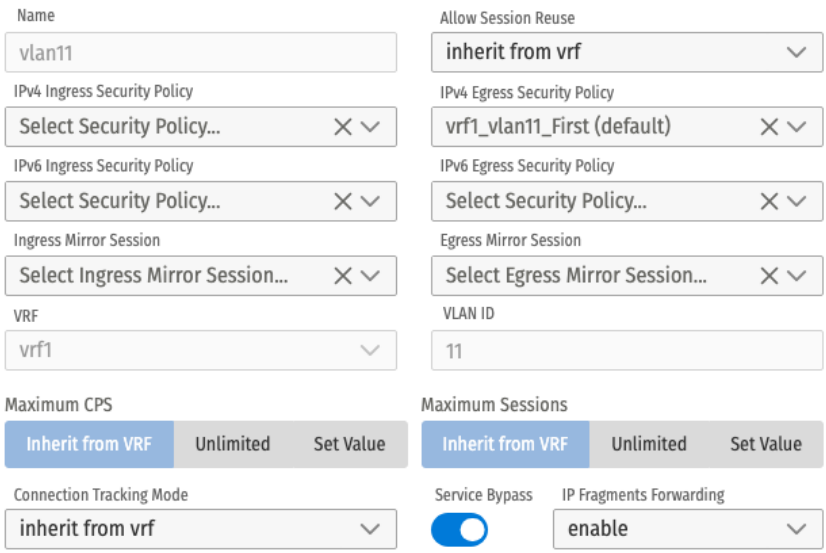
Any firewall by default will drop fragmented packets, In PSM you can enable support of fragmentation to allow fragmented packets. This is helpful in scenarios such as the early stages of deployment while you are still gaining visibility and learning about what is going on in the network.

To test this, you will send Jumbo ping packets before and after you set this flag to validate the feature.

Step 1 - From your **VM 1** #ping (workload2-ip) -s 9000

- What happens?

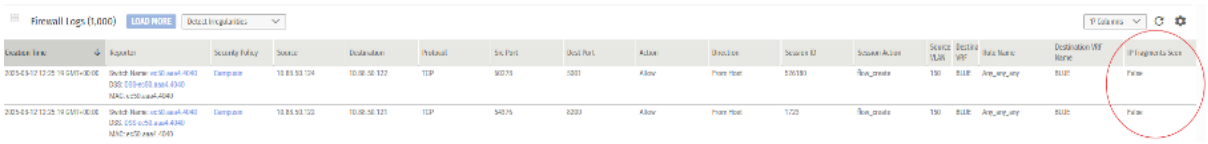
Step 2 - From the **Networks** section go to your vlan and enable IP fragment forwarding in the drop-down menu, click save and ok on pop-up.




Step 3 - From your VM 1 #ping (workload2-ip) -s 9000

- What happens?

Now check the logs in both PSM and ELK and notice the fragmentation flag is set.



Learning Check Task 3

1. Using the PSM GUI to verify a policy
2. Working with the IP Fragments feature
3. Verify ping and log visibility

## Task 4 - PSM Connection Tracking (CT)

DSM modules enforce stateful firewall policies. However, when transitioning from no policy to full enforcement, starting in stateless mode can help. It lets you drop traffic with minimal disruption while monitoring network behavior.

To use stateless mode, set the connection tracking state to 'Disabled' instead of the default 'Enabled'. In this mode, rules act like traditional ACLs, so you may need to define traffic in both directions for bidirectional communication.

**Before making any changes to stateless mode:**

1. Enable service bypass from your vlan and create a new rule from your security policy to block SSH. (apply on egress on the network). (click Save)

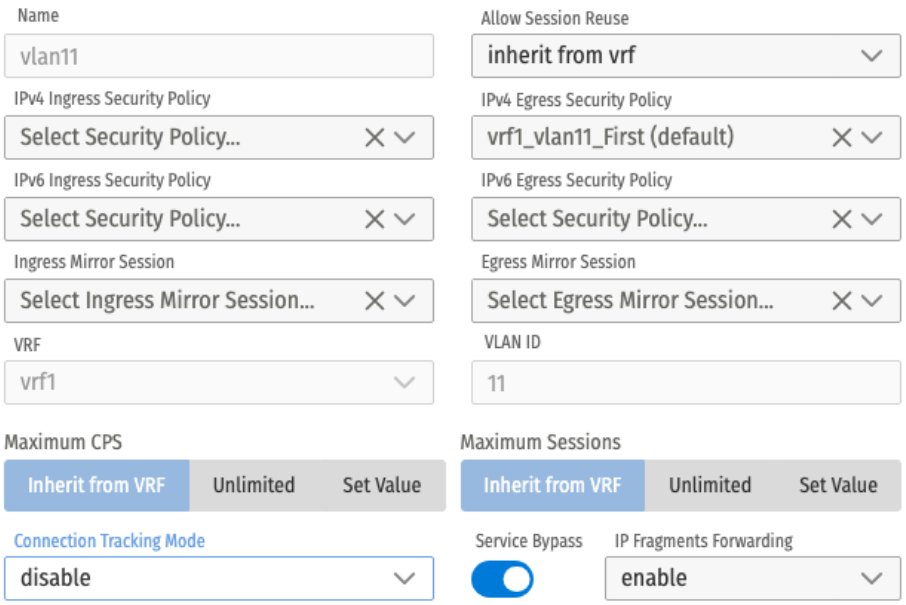
2. Connect a SSH session between workload 1 and workload 2.

| User | Password |
|---|---|
| tc | VMware1! |

#ssh tc@192.168.x.x

3. With connection tracking enabled, disable service bypass, what happens to the SSH session. (Save)
4. Re-enable service bypass. (Save)

**Change to stateless. (disable connection tracking).**




5. Connect a SSH session between workload 1 and workload 2.
6. With connection tracking disabled, disable service bypass, what happens to the SSH session. (Save)
7. Re-enable service bypass. (Save)

## Learning Check Task 4

1. Using the PSM GUI to verify a policy
2. Working with the Service Bypass feature
3. Working with the Connection Tracking feature
4. Testing SSH connectivity

## Task 5 - Block IPERF traffic based on IP-Collection

An IP collection is a list of IP addresses that can be referenced at a later time in a firewall policy. This makes policy creation simpler and allows for easier changes to policy in the future. We will add IP Collections for your selected POD Networks.

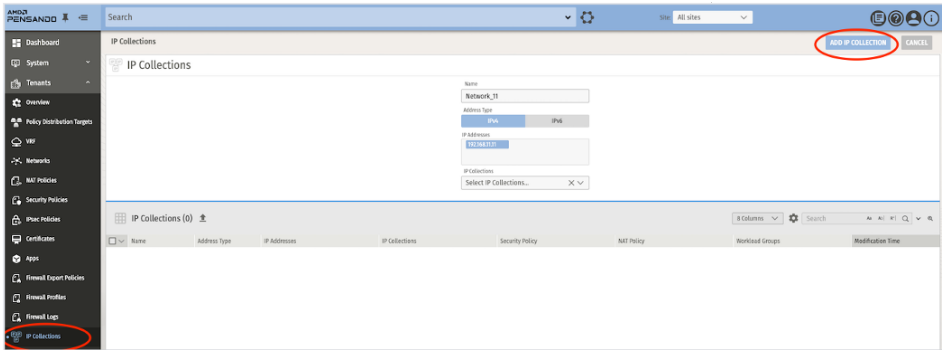
From your PSM:

In this scenario, the IP Collections have already been preconfigured on the PSM instance for each specific POD, based on its associated VRF and VLAN. If you were to create them manually, you would follow the steps outlined below.

In the **PSM** GUI move to the **Tenants - IP Collections –Add IP collection** section.

- Provide a name for the Network = **Network_ XX**
- Add the correct IP Addresses of your workloads = **192.168.xx.xx**
- Leave “IP Collections” field blank
- Submit by clicking “Add IP Collection” at top of page
- Click “Save Policy” at the top to deploy the changes

Repeat for additional Networks




iPerf is a tool for active measurements of the maximum achievable bandwidth and throughput (UDP/TCP) on IP networks between a sender and a receiver as well as other abilities.

To test this, we will use 2 hosts **WS1 & WS2** in your respective environment and one will be used as a sender and the other as a receiver.

We will now configure a simple deny IPERF policy between your workload1 (Server) and workload2 (client) using an IP collection to test this specific feature.

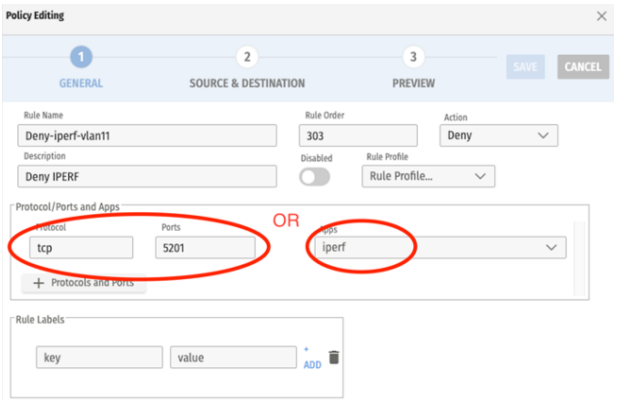
Before creating any policy:

- Proceed to test if you can start the iperf server on WS1 and initiate an iperf connection on port 5201 from WS2. (Reference the commands in the “from your Host” section below and run them from your WorkStation)
- Review the logs in PSM and ELK
- Do they match what you expect?

From your PSM:
In the **PSM** GUI move to the **Tenants - Security Policy** section. Hover over the existing policy named <vrfx_vlanxx_first> and select the pencil on the right side to “Edit Policy on Details Page”. Click the “+” to Add a rule to our policy.

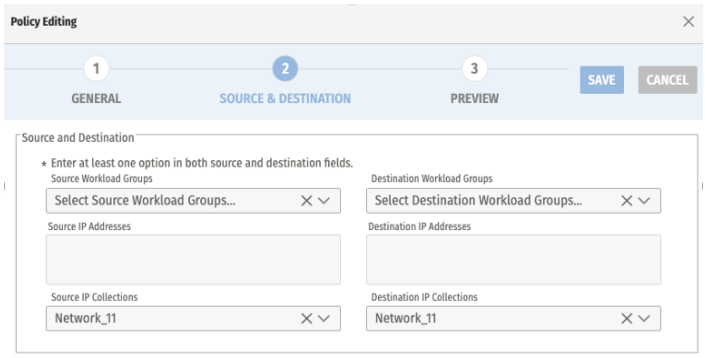
Step 1

- Provide a name for this rule “Deny-iperf-vlan x” (vlan x is one of your choosing from your pod)
- Provide an Action of type “Deny”
- Set Rule Order to 1
- Set App to **iPerf** or type the protocol tcp/port 5201 in the appropriate box




Step 2

- Proceed to section 2 for “Source and Destination”
- In the drop down change the Destination and Source IP Collections to be the statically configured Network_xx ip collection= created above




Step 3

- Preview the new Rule in “vrfx_vlanxx_first”
- Click Save
- Click “Save Policy” at the top to deploy the change.

After policy creation:

- Proceed to test if you can start the iperf server on WS1 and initiate an iperf connection on port 5201 from WS2. (Reference the commands in the “from your Host” section below and run them from your WorkStation)
- Review the logs in PSM and ELK
- Do they match what you expect?

From your hosts:

iperf3 is already installed on the workstations and you can verify this by running the following command from the VM’s console-
#iperf3 –v

⚠️ Iperf3 listens on port 5201 by default

To test
From **Workstation** #1 we will set up iperf server to run and start listening on port 5201 with the following command:
**#**iperf3 -s -p 5201

From **Workstation** #2 we will initiate an iperf connection on port 5201 to see if the connection is allowed with the following command:

# iperf3 -c 192.168.x.x -t 10 -b 1000000000 -p 5201

⚠️ If there are any issues with starting iperf on the server machine such as ‘Unable to start listener…’ you can run the following commands and try again:

#ps –ef
Find the pid number for the iperf process
#sudo kill *PID*

Learning Check Task 5

1. Using the PSM GUI to verify policy
2. Verifying IP collections
3. Adding a rule to existing policy for iperf
4. Testing host to host connectivity
5. Verifying log visibility

## Task 6 - Block SSH traffic based on VMware Tag's

A *workload group* is a logical grouping of endpoint IP addresses or workload objects based on either one or more workload labels or tags, a list of IP collections, or a mix of both workload labels and IP collection objects. A workload-based label selector can be used only on those workloads/VMs that are dynamically imported from vCenter, with the assumption that the workloads are already tagged in vCenter prior to import.

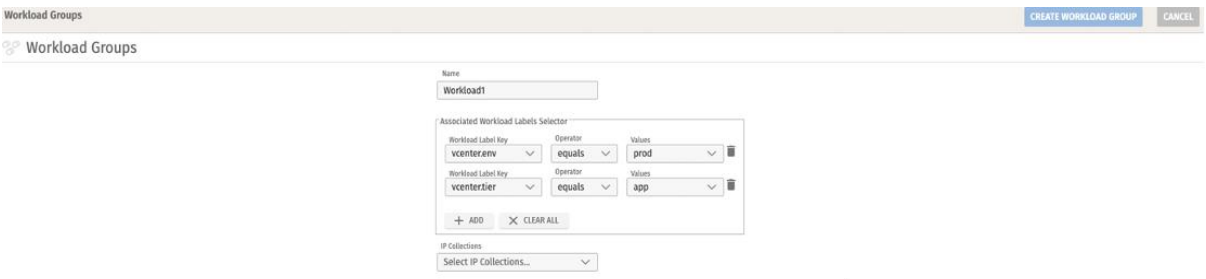
When workload groups are used in a security policy, it enhances the user experience of defining a policy and brings out the abstraction of IP address constructs, as policy no longer needs to be based on IP addresses.

From your PSM:

In this scenario, the Workload Groups have already been preconfigured on the PSM instance for each specific POD, based on its associated labels and values. If you were to create them manually, you would follow the steps outlined below.

In the **PSM** GUI move to the **Workload – Workload Groups – Add Workload Group** section.

- Provide a name for the Workload Group **Workload_x**
- Select the desired Workload Label Key **vcenter.env**
- Select the operator as **equals**
- Select the value as **prod**
- Select the desired Workload Label Key **vcenter.tier**
- Select the operator as **equals**
- Select the value as **app**
- Click "Save Policy" at the top to deploy the changes.




In your vSphere environment the VM's already contain the appropriate Tags and attributes …

SSH

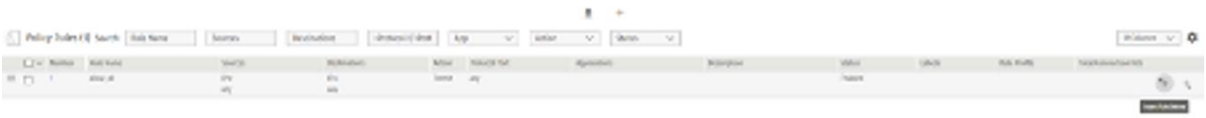
We will now configure a simple deny SSH policy between your workload1 and workload2 (directional) to test this specific feature. To test this, again we will use 2 hosts WS1 & WS2 in your respective environment.

Before creating any policy:

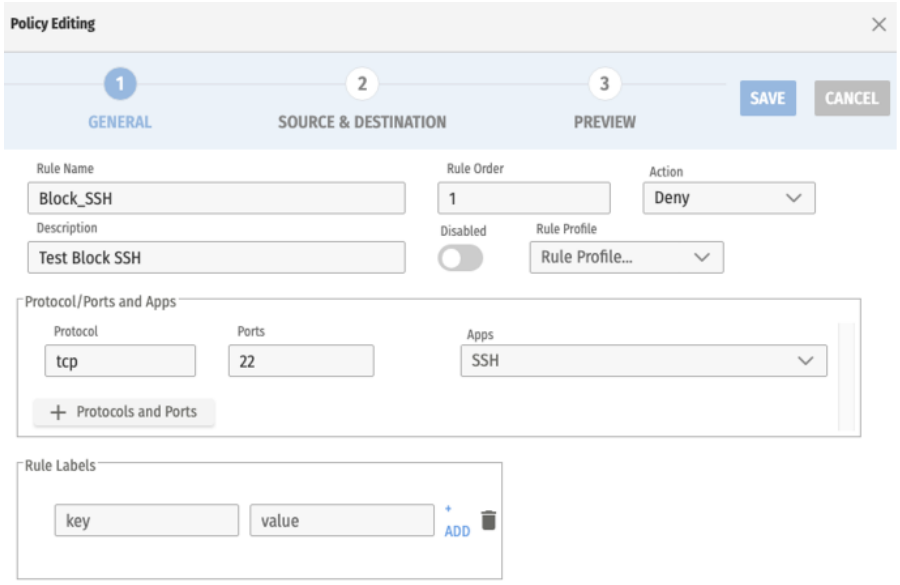
- Proceed to test if you can ssh from host 1-2 and 2-1. Also review the logs in PSM and ELK
- Do they match what you expect?
- Did you verify service bypass setting?

From your PSM:

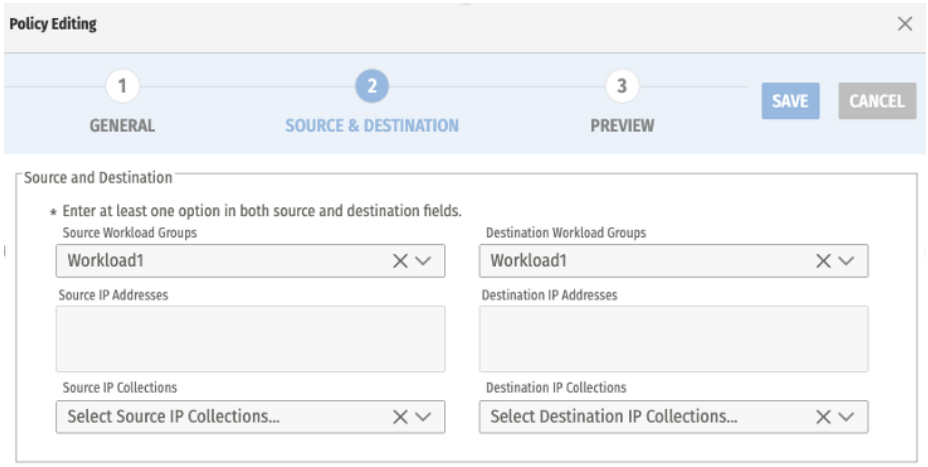
In the **PSM** GUI move to the **Tenants - Security Policy** section. Hover over the existing policy named <vrfx_vlanxx_first> and select the pencil on the right side to "Edit Policy on Details Page". Click the "+" to Add a rule to our policy. Under policy Rules click the "+" to add a rule to your policy.



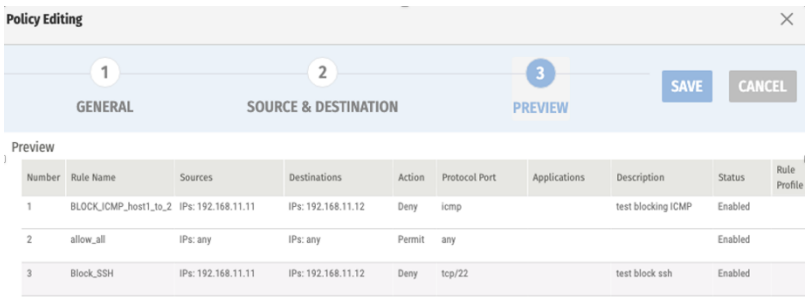
Step 1- You need to provide a "Name" for the rule, verify the rule order is above the default_allow, change the action to "Deny", add a description if needed and type the protocol tcp/port 22 in the appropriate box. Enable rule is default




Step 2 - In the policy editing window from the Source Workload Groups dropdown select the previously created workload group




Step 3 - Preview the rule to make sure it meets the requirement and click save and Save policy at top of screen.




- Proceed to test if you can ssh from host 1-2 and 2-1. Also review the logs in PSM and ELK
- Do they match what you expect?

- Did you check service bypass setting?

 Learning Check Task 6

1. Using the PSM GUI to verify policy
2. Verifying Workload groups
3. Adding a rule to existing policy for iperf
4. Testing host to host connectivity
5. Verifying log visibility

## Task 7 - Move workload to separate Networks and subnets (Working on update)

Possible front end to back-end scenario

Vlan x to y

NGINX scenario w / Curl

Tests inter VLAN egress and ingress rules.
Test with workload groups and IP collections.

# Summary

This guide offered a high-level overview of the CX 10000's capabilities. It was designed to help you explore and experiment with vSphere, PSM and ELK in a controlled lab environment. The goal was to get you familiar with what you can configure and manage through these tools. Hopefully this provided you the opportunity to test, learn, and deepen your understanding of the entire solution. Feel free to reach out with any issues, questions, additional use-cases or recommended improvement and keep an eye out for the survey that will be sent out after the lab completion.