

# Lab 06 - L7 Security - L7 FW, FQDN Filtering & IDPS

## Introduction

VMware Cloud on AWS provides VMware's enterprise class SDDC software on AWS cloud. It includes a robust set of networking and security capabilities that enable customers to run production applications in the cloud. Every SDDC is provisioned with the Gateway Firewall to protect the perimeter of the SDDC, and the Distributed Firewall to secure lateral communication across workloads inside the SDDC. Powered by the proven security capabilities of VMware NSX-T, Gateway and Distributed Firewall provide enterprise class Layer 4 security for applications in VMware Cloud on AWS:

- Gateway Firewall enables customers to selectively allow and deny traffic from and to applications deployed in the SDDC. It also controls access to management infrastructure, such as vCenter and NSX manager
- Distributed Firewall is built into the hypervisor and automatically scales across every host in the SDDC. Enabling micro-segmentation at the workload level, Distributed Firewall policies migrate with the VM when they move from host to host in the SDDC.

NSX Advanced Firewall features take the network security capabilities of VMware Cloud on AWS SDDC to the next level, allowing customers to define security policies at Layer 7 and enabling deep packet inspection across all vNICs within the SDDC.



#### Distributed IDS/ IPS

Distributed traffic inspection that scales seamlessly, with context based threat prevention.



#### Distributed Firewall with Layer 7 Application ID

Deep Packet Inspection built into the hypervisor with built in profiles for common enterprise applications.



#### Distributed Firewall with Active Directory based User ID

Per user and session application access control with an Identity Firewall.



#### Distributed Firewall with FQDN Filtering

Permit or deny communication to specific destinations in the Internet.

With the NSX Advanced firewall add-on to your VMC on AWS SDDC(s) you can deliver enhanced security for your VMC on AWS workloads in any of these scenarios:

- **Distributed IDS/IPS** (Detect and prevent threats to your workloads) - Enterprises are constantly reminded of threats to their applications by a never-ending stream of news about exploits on the internet. With NSX Distributed IDS/ IPS, customers gain protection against attempts to exploit vulnerabilities in workloads on VMware Cloud on AWS. Distributed IDS/ IPS is an application-aware deep packet inspection engine that can examine and protect traffic inside the SDDC. Customers can detect and prevent lateral threat movement within the SDDC using the intrinsic security capabilities of Distributed IDS/IPS.
- **L7 (Context-aware) Firewall** - With L7 (Context-aware) firewall you can go beyond simple IP/ port level layer 4 security to complete stateful layer 7 controls and filtering. Deep packet inspection (DPI) built into the Distributed Firewall enables you to allow only the intended application / protocols to run, while denying all other traffic at the source. This enables you to isolate sensitive applications by creating virtual zones within the SDDC. Distributed Firewall (DFW) layer 7 policies are enforced at the hypervisor (vNIC) level and can migrate with the VM when they move from host to host in the SDDC, ensuring there are no gaps in enforcement.
- **User Identity Firewall (IDFW)** - You can create groups based on User ID and define DFW rules to control access to virtual desktops and applications in the SDDC. Per user/ user session access control limits the amount of time and exposure users have to desktops or applications. Integration with Active Directory / LDAP enables the DFW to continuously curate user access to applications. User ID based rules are enforced by the DFW at the source, delivering pervasive, intrinsic security throughout the SDDC.


- FQDN Filtering - Applications that communicate outside the SDDC also gain layer 7 protection using Distributed Firewall FQDN filtering capability. Customers can define specific FQDNs that are denied access to applications in the SDDC. The DFW maintains the context of VMs when they migrate. Customers increasingly rely on application profiling and FQDN filtering to reduce the attack surface of their applications to designated protocols and destinations.

## TASKS

### Task 1 - Enable the NSX Advanced Firewall Add-on

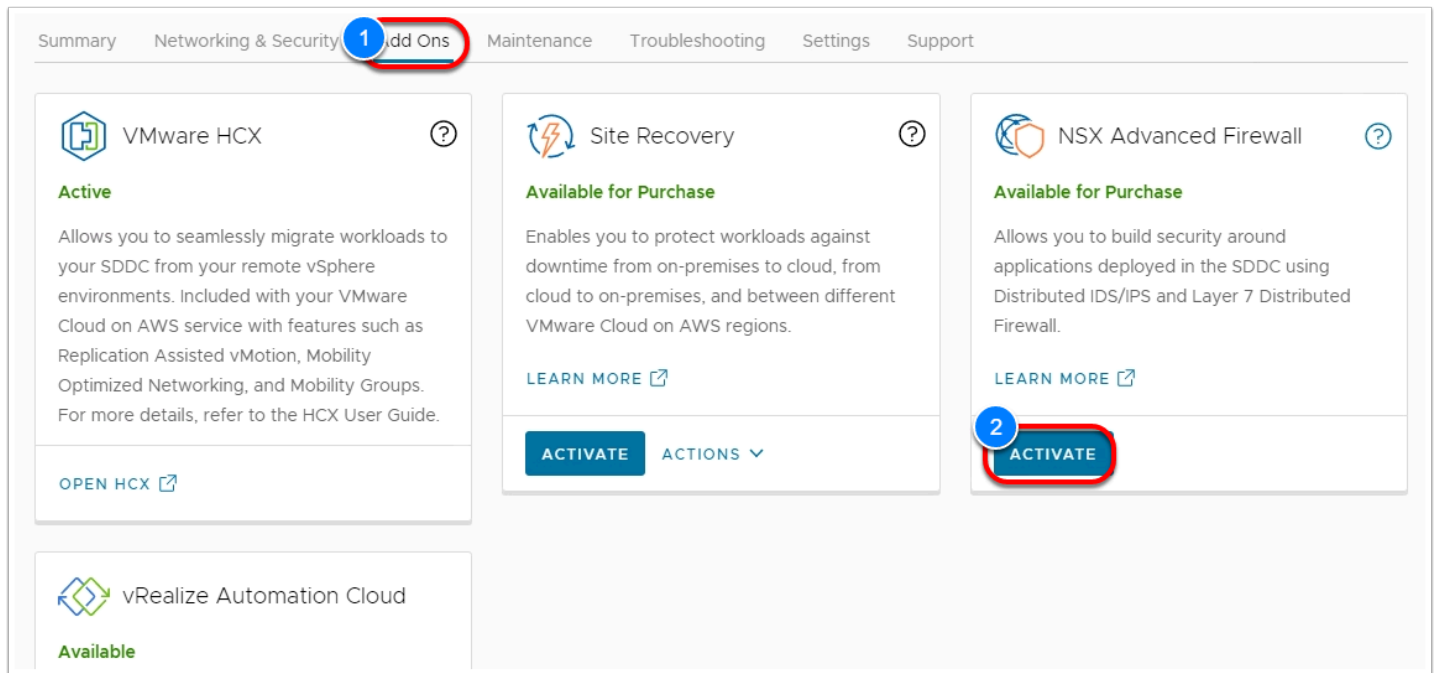
-  The NSX Advanced Firewall Add-on adds Layer-7 Firewall protection, Identity Firewalling, Distributed IDS/IPS and FQDN Filtering to the VMC on AWS SDDC. This Feature is an Add-on featured and priced in addition to the Standard VMC on AWS subscription.

Before any of these features can be used, you must first enable the add-on onto your SDDC. In this tasks, we'll walk through the steps of enabling the NSX Advanced Firewall functionality onto your SDDC.

-  **NOTE:** This feature isn't readily accessible to customers with an existing SDDC(s). Customers will need to request access to this feature and have their SDDC(s) upgraded to the M15 release to take advantage of this feature. Later this year, at the release of M16 customers will no longer have to request the feature.

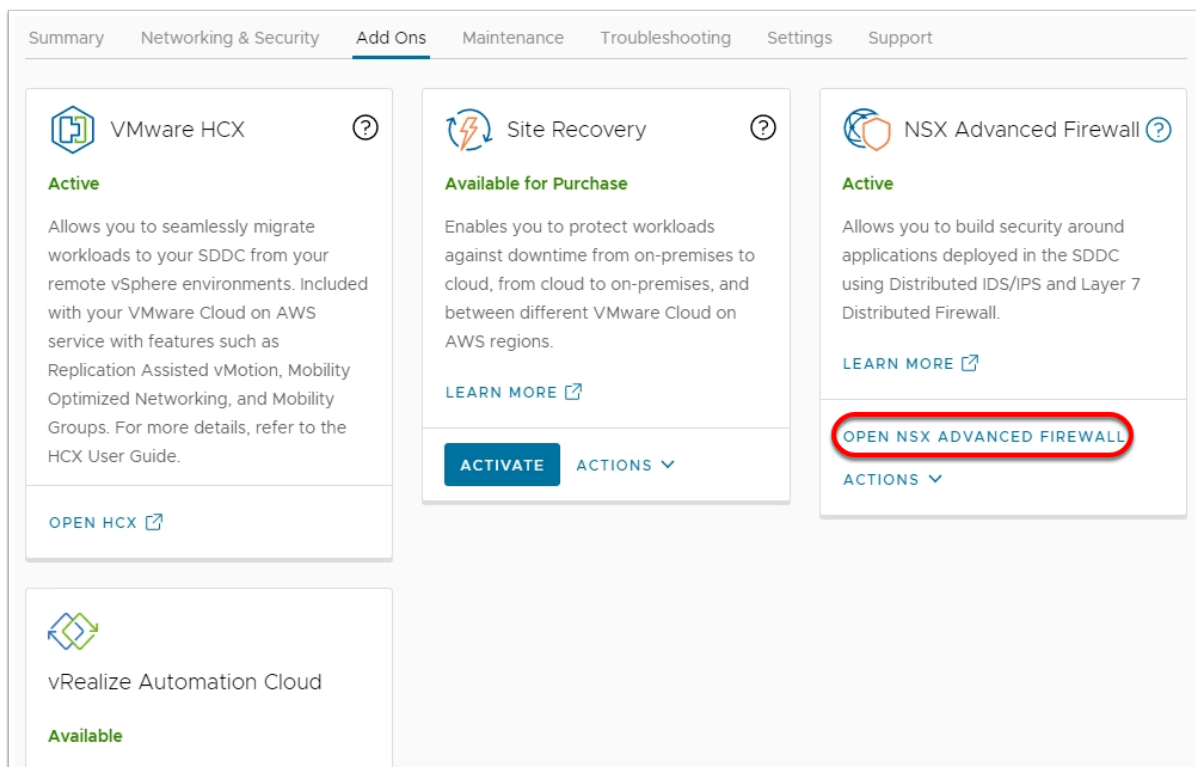
1. From the VDI Desktop open the **Google Chrome** Browser or **Firefox**
2. Launch the **VMware Cloud SDDC** bookmark or browse to <https://vmc.vmware.com/console/sddc>
3. Login as
  - **User name:** **vmcexpert#-xx@vmware-hol.com** (Where **#** is your Environment ID and **xx** is your student number)
  - **Password:** **VMware1!**
4. On your SDDC tile, click **View Details**
5. Click the **Add-Ons** tab


6. In the NSX **Advanced Firewall Tile**, click **Activate**



7. Click **Activate**

8. Click **OPEN NSX ADVANCED FIREWALL** (This will take you to the Networking & Security Tab)



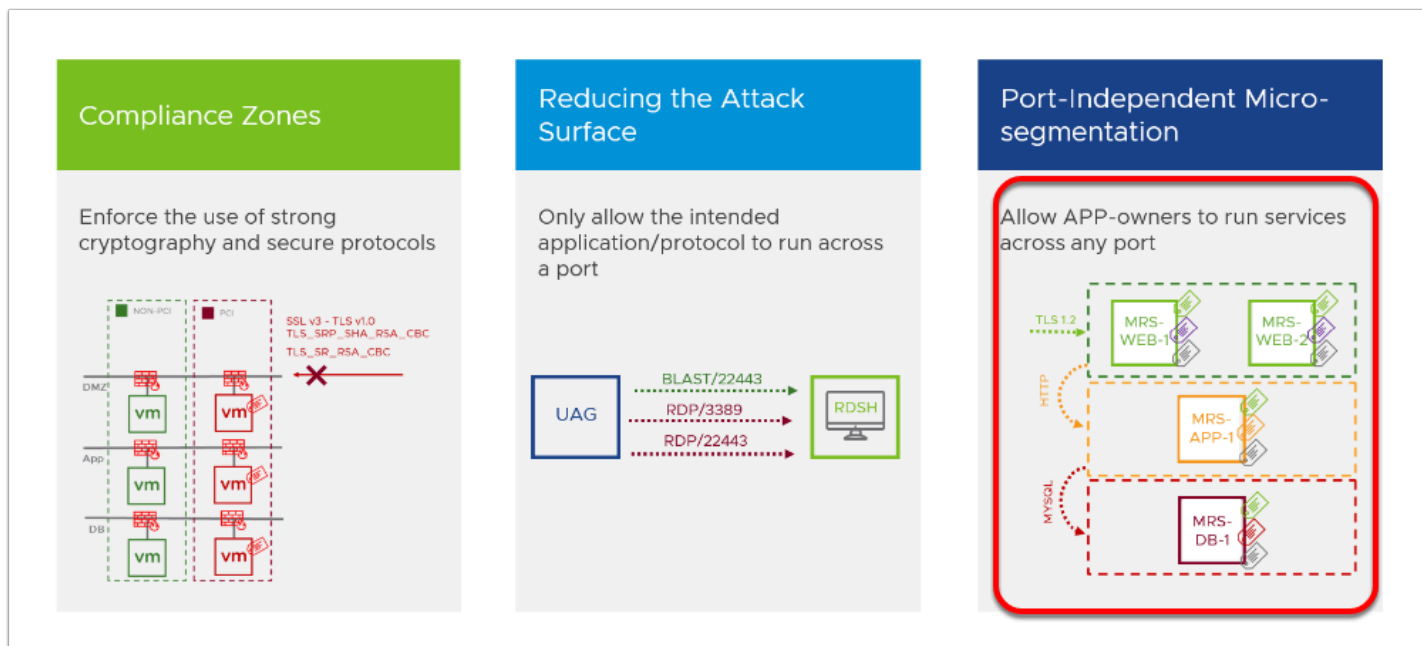
 At this point The NSX Advanced Firewall Add-on has been enabled. To make use of the functionality it provides, you must configure them individually. In the upcoming tasks we will configure and test each of these features.

## Task 2 - Configure a Context-Aware Firewall rule.

NSX Context-Aware Firewall Rule (L7) enhances visibility at the application level and helps to override the problem of application permeability. Visibility at the application layer helps you to monitor the workloads better from a resource, compliance, and security point of view.

Firewall rules cannot consume application IDs. Context-aware firewall identifies applications and enforces a micro-segmentation for EAST-WEST traffic, independent of the port that the application uses. Context-aware or application-based firewall rules can be defined by defining Layer 7 service objects. After defining Layer 7 service objects in rules, you can define rules with specific protocol, ports, and their application definition. Rule definition can be based on more than 5-tuples. You can also use Application Rule Manager to create context-aware firewall rules.

With Context-Aware Firewalling you can enable enforcement of security protocol versions/ ciphers reduce attacks by only allowing traffic matching APP Fingerprint, and enforce port-independent rules

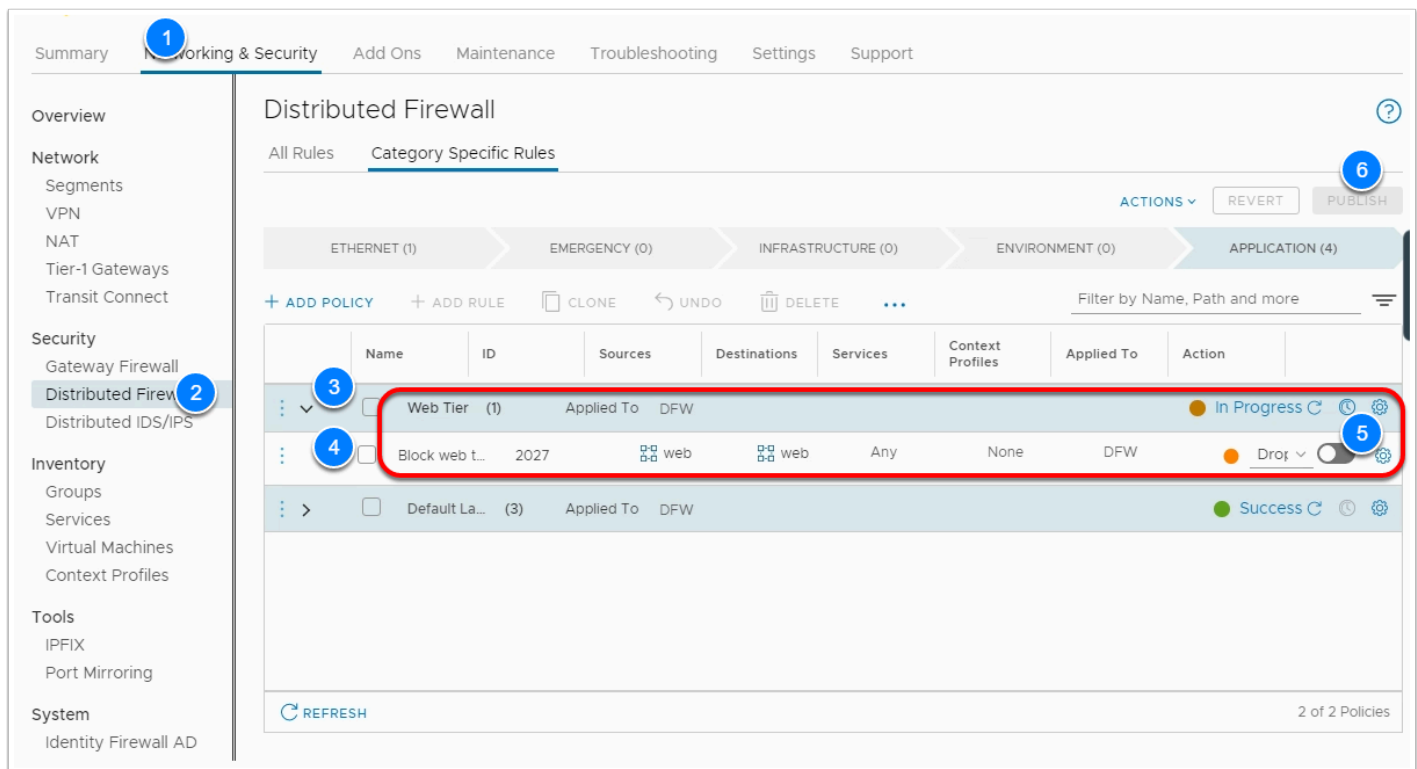


## Task 2.1 - Create an L4 Firewall rule for SSH

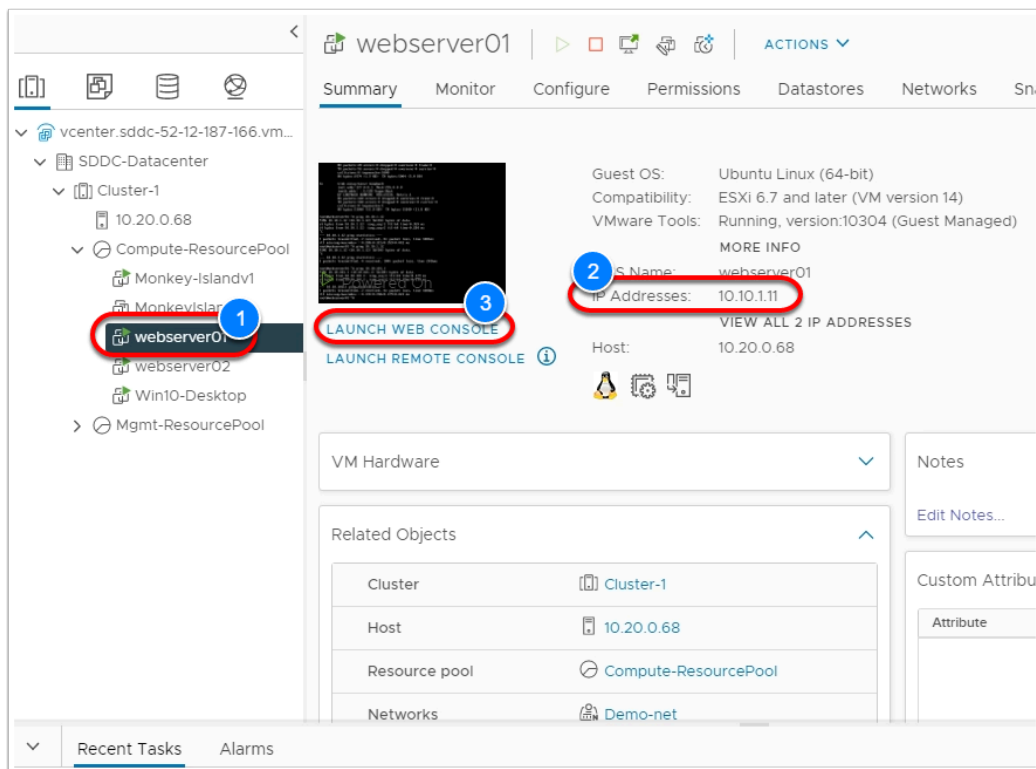
We will begin by creating a standard L4 Distributed firewall rule for SSH, doing so will allow us to better understand the power L7 rules bring to securing applications running in your SDDC.

1. Click The Networking and Security tab of your SDDC
2. Click **Distributed Firewall**
3. Click on **Actions** and choose **General Settings**
4. Click the slider for **Identity Firewall Status** to enable it
5. Click **Identity Firewall Settings**
6. Click the slider for **Cluster-1** to enable it
7. Click **SAVE**
8. Expand the **Web-Tier** Firewall Policy
9. At the far right of the **Block Web-to-web** rule
10. Move the **slider** to the left temporarily disable the rule.
11. Click **PUBLISH**

**NOTE:** Doing this will allow communications between the web servers. We created this rule in lab 2 and set it to block all traffic between the web servers.



12. From the Settings tab Open the SDDC vCenter (if you no longer have it opened)
13. Log into the SDDC vCenter as:
  - **cloudadmin@vmc.local**
  - **<copy the cloudadmin password from the settings tab or your worksheet>**
14. In the vCenter Inventory, select **webserver02** and take note of its IP address (**10.10.X.X**)
15. Select **webserver01** take note of it's IP address, and click **LAUNCH WEB CONSOLE**




16. In the browser tab for webserver01, login as:

- **root**
- **VMware1!**

17. attempt an ssh session to webserver02

```
<p>ssh <webserver02_ipaddress></webserver02_ipaddress></p>
```

 Click to copy

```
Ubuntu 16.04.5 LTS webserver01 tty1

Hint: Num Lock on

webserver01 login: root
Password:
Last login: Sun Jul 25 19:12:18 EDT 2021 on tty1
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

255 packages can be updated.
157 updates are security updates.

root@webserver01:~# ssh 10.10.1.12
root@10.10.1.12's password:
```

18. If prompted with the SSH Thumbprint type **Yes**, then Press **enter**.  
You should be prompted to log into webserver02
19. Press **ctrl+c** on the keyboard to exit SSH
20. Back in the SDDC console, let's modify the firewall rule to block SSH between the web servers
21. In the Services field of the **Block web-to-web** rule, move your mouse to the right of **ANY**
22. click the **blue pencil** that appears
23. In the dialog find and select **SSH**
24. Click **APPLY**
25. In the far right **move the slider** to the right to **enable** the rule
26. Click **PUBLISH**

Summary **Networking & Security** Add Ons Maintenance Troubleshooting Settings Support

Overview

Network

- Segments
- VPN
- NAT
- Tier-1 Gateways
- Transit Connect

Security

- Gateway Firewall
- Distributed Firewall**
- Distributed IDS/IPS

Inventory

- Groups
- Services
- Virtual Machines
- Context Profiles

Tools

- IPFIX
- Port Mirroring

System

## Distributed Firewall

All Rules Category Specific Rules

1 Total Unpublished Change ACTIONS REVERT PUBLISH

ETHERNET (1) EMERGENCY (0) INFRASTRUCTURE (0) ENVIRONMENT (0) APPLICATION (4)

+ ADD POLICY + ADD RULE CLONE UNDO DELETE ... Filter by Name, Path and more

1 Unpublished Change

	Name	ID	Sources	Destinations	Services	Context Profiles	Applied To	Action
⋮	Web Tier (1)		Applied To	DFW				In Progress
⋮	Block web t...	2027	web	web	Any	None	DFW	Drop
⋮	Default La...	(3)	Applied To	DFW				Success

REFRESH 2 of 2 Policies

Rule > Block web to Web

Services (1) Raw Port-Protocols (0)

SSH X

SSH X 1 CLEAR X

	Name	Service Entries	Status
⋮	SSH	TCP (Source: Any   Destination: 22)	Success

1 1 - 1 of 1 Services

Show Only Selected

CANCEL APPLY

Distributed Firewall

All Rules Category Specific Rules

1 Total Unpublished Change ACTIONS REVERT PUBLISH 5

ETHERNET (1) EMERGENCY (0) INFRASTRUCTURE (0) ENVIRONMENT (0) APPLICATION (4)

+ ADD POLICY + ADD RULE CLONE UNDO DELETE ... Filter by Name, Path and more

1 Unpublished Change

	Name	ID	Sources	Destinations	Services	Context Profiles	Applied To	Action
⋮	Web Tier (1)		Applied To	DFW				In Progress 4
⋮	Block web t...	2027	web	web	SSH	None	DFW	Drop 5
⋮	Default La...	(3)	Applied To	DFW				Success

27. Return to the console tab for webserver01 and once more attempt an SSH session to webserver02

28. You can use the Up Arrow key to recall the previous command

```
<p>ssh <webserver02_ipaddress></webserver02_ipaddress></p>
```

Click to copy


```
Ubuntu 16.04.5 LTS webserver01 tty1
Hint: Num Lock on

webserver01 login: root
Password:
Last login: Sun Jul 25 19:12:18 EDT 2021 on tty1
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)


 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

255 packages can be updated.
157 updates are security updates.

root@webserver01:~# ssh 10.10.1.12
root@10.10.1.12's password:
root@webserver01:~# ssh 10.10.1.12
ssh: connect to host 10.10.1.12 port 22: Connection timed out
root@webserver01:~#
```

 This time notice that the connection timed out and was never accepted by webserver02, this is because the DFW blocked the session and it never made its way to webserver02

## Task 2.2 - Create an L7 Context-aware Rule for SSH

 In task 2.1, we created a standard L4 rule to block SSH traffic between the web servers and it worked as expected. This is because, at L4 the DFW evaluates the Source, Destination, and Port of the packet. In this case, the source was webserver01, the destination was webserver02 and the destination port was 22. What if SSH was listening on another port, however? What if some nefarious person (knowing SSH on port 22 is being blocked) changed the port the server listens on and attempts to SSH to the server against this new port, what happens then? Let's find out in this task and also see how an L7 rule can protect against this type of activity.

1. In the vCenter Inventory, select **webserver02**
2. Click **LAUNCH WEB CONSOLE**
3. In the browser tab for webserver01, login as:
  - **root**

- **VMware1!**

**i** We will now modify SSH on webserver02 to listen on port 2222 instead of 22, but before doing so, we will create a backup of the configuration file.

4. Type the following commands in the console of webserver02 to create a copy of the ssh configuration file

```
<p>cd /etc/ssh  
ls  
cp sshd_config sshd_config.orig  
ls</p>
```

 Click to copy

```
Ubuntu 16.04.5 LTS webserver02 tty1  
Hint: Num Lock on  
webserver02 login: root  
Password:  
Last login: Sun Jul 25 19:47:10 EDT 2021 on tty1  
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
255 packages can be updated.  
157 updates are security updates.  
  
root@webserver02:~# cd /etc/ssh 1  
root@webserver02:/etc/ssh# ls 2  
moduli      ssh_host_dsa_key      ssh_host_ecdsa_key.pub  ssh_host_rsa_key  
ssh_config  ssh_host_dsa_key.pub  ssh_host_ed25519_key    ssh_host_rsa_key.pub  
sshd_config ssh_host_ecdsa_key    ssh_host_ed25519_key.pub ssh_import_id  
root@webserver02:/etc/ssh#  
root@webserver02:/etc/ssh# cp sshd_config sshd_config.orig 3  
root@webserver02:/etc/ssh# ls 4  
moduli      sshd_config.orig      ssh_host_ecdsa_key      ssh_host_ed25519_key.pub  ssh_import_id  
ssh_config  ssh_host_dsa_key      ssh_host_ecdsa_key.pub  ssh_host_rsa_key  
sshd_config ssh_host_dsa_key.pub  ssh_host_ed25519_key    ssh_host_rsa_key.pub  
root@webserver02:/etc/ssh#
```

**i** Now, let's edit the configuration file to set the servers SSH port to 2222

5. Type **sudo nano sshd\_config** to open the configuration file in nano
6. look for the line with Port 22 and change the port to **2222**
7. Press **ctrl+O** to save the file
8. Press **enter** to confirm the save
9. Press **ctrl+X** to exit nano

```
# Package generated configuration file
# See the sshd_config(5) manpage for details

# What ports, protocols and protocols we listen for
Port 2222
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 1024

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 120
PermitRootLogin prohibit-password
StrictModes yes

RSAAuthentication yes
PubkeyAuthentication yes
```

1

2

3

10. restart the ssh service by typing the following command

```
<p>sudo systemctl restart sshd.service</p>
```

Click to copy

11. Return to the console tab for webserver01 and once more attempt an SSH session to webserver02 but this time on port 2222 by typing the following command

```
<p>ssh -p 2222 <webserver02_ipaddress></webserver02_ipaddress></p>
```

Click to copy

```
root@webserver01:~#
root@webserver01:~# ssh -p 2222 10.10.1.12
root@10.10.1.12's password: _
```

Press **ctrl+c** to exit the prompt

**i** This time the connection does not timeout, the DFW doesn't block it. As mentioned earlier the firewall is looking for SSH on port 22, not port 2222, so we can bypass the firewall policy. we will now see what happens when we apply context awareness to the firewall rule.

12. In the VMC on AWS SDDC Console Click the **Networking and Security** tab
13. Click **Distributed Firewall**
14. Expand the **Web Tier** Policy
15. In the Services field mouse over **SSH** and click the **blue pencil**
16. In the dialog remove **SSH** from the list of selected service
17. Click **APPLY**
18. In the Context Profile field, mouse over **None** and click the **blue pencil**
19. Select **SSH**
20. Click **APPLY**
21. Click **PUBLISH**

Select Context Profile

Rule > Block web to Web

SSH X

ADD CONTEXT PROFILE

	Name	Attributes	Description	Tags	Where Used	Status
<input type="checkbox"/>	SNMP	SNMP	SNMP (Simple Network Management Protocol) is an Internet-standard protocol for managing devices on IP networks.	0	0	Success
<input checked="" type="checkbox"/>	SSH	SSH	SSH (Secure Shell) is a network protocol that allows data to be exchanged using a secure channel between two networked devices.	0	1	Success
<input type="checkbox"/>	SSL	SSL	SSL (Secure Sockets Layer) is a cryptographic protocol that provides security over the Internet.	0	0	Success
<input type="checkbox"/>	SYMUPDAT	SYMUPDAT	Symantec LiveUpdate traffic	0	0	Success

☒ 1 REFRESH

K < 2 / 2 > | 51 - 63 of 63 items

Show Only Selected

CANCEL APPLY

Summary Networking **Security** Add Ons Maintenance Troubleshooting Settings Support

Overview

Network

- Segments
- VPN
- NAT
- Tier-1 Gateways
- Transit Connect

Security

- Gateway Firewall
- Distributed Firewall**
- Distributed IDS/IPS

Inventory

- Groups
- Services
- Virtual Machines
- Context Profiles

Tools

- IPFIX
- Port Mirroring

System

- Identity Firewall AD

## Distributed Firewall

All Rules Category Specific Rules

1 Total Unpublished Change ACTIONS REVERT PUBLISH

ETHERNET (1) EMERGENCY (0) INFRASTRUCTURE (0) ENVIRONMENT (0) APPLICATION (4)

+ ADD POLICY + ADD RULE CLONE UNDO DELETE ... Filter by Name, Path and more

1 Unpublished Change

Name	ID	Sources	Destinations	Services	Context Profiles	Applied To	Action
Web Tier (1)		Applied To	DFW			In Progress	
Block web t...	2027	web	web	Any	SSH	DFW	Drop
Default La...	(3)	Applied To	DFW			Success	
Default Rul...	3	Any	Any	IPv6...	None	DFW	Allow
Default Rul...	4	Any	Any	DHC...	None	DFW	Allow

REFRESH 2 of 2 Policies

22. Return to the console tab for webserver01 and once more attempt an SSH session to webserver02 on both ports 22 and 2222

```
<p>ssh <webserver02_ipaddress>
ssh -p 2222 <webserver02_ipaddress></webserver02_ipaddress></webserver02_ipaddress></p>
```

Click to copy

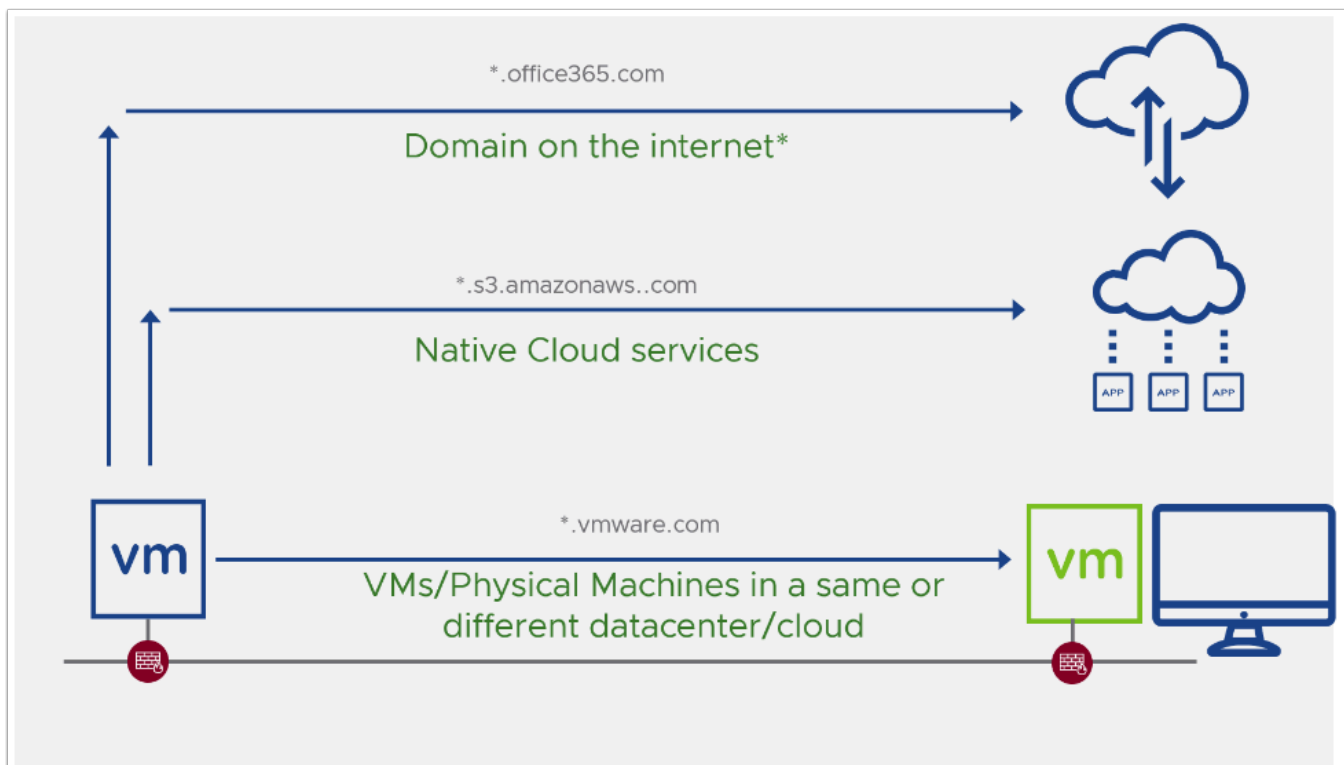
```
root@webserver01:~#
root@webserver01:~#
root@webserver01:~# ssh 10.10.1.12
ssh: connect to host 10.10.1.12 port 22: Connection refused
root@webserver01:~#
root@webserver01:~#
root@webserver01:~# ssh -p 2222 10.10.1.12
packet_write_wait: Connection to 10.10.1.12 port 2222: Broken pipe
root@webserver01:~# _
```

- i** This time both attempts (on the standard ssh port of 22 and the modified port of 2222) is not allowed. This is because the DFW now assesses the packet at layer 7 and identifies the heuristics of the packet to be ssh and does not allow the traffic through.

## Task 3 - FQDN Filtering

VMC on AWS can allow users to only access specific domains by whitelisting and/or blacklisting FQDNs. In many high-security environments, outgoing traffic is filtered using the Distributed firewall. When you want to access an external service, you usually create IP-based firewall rules. In some cases, you don't know which IP addresses hide behind a domain. This is where domain filters come in handy.

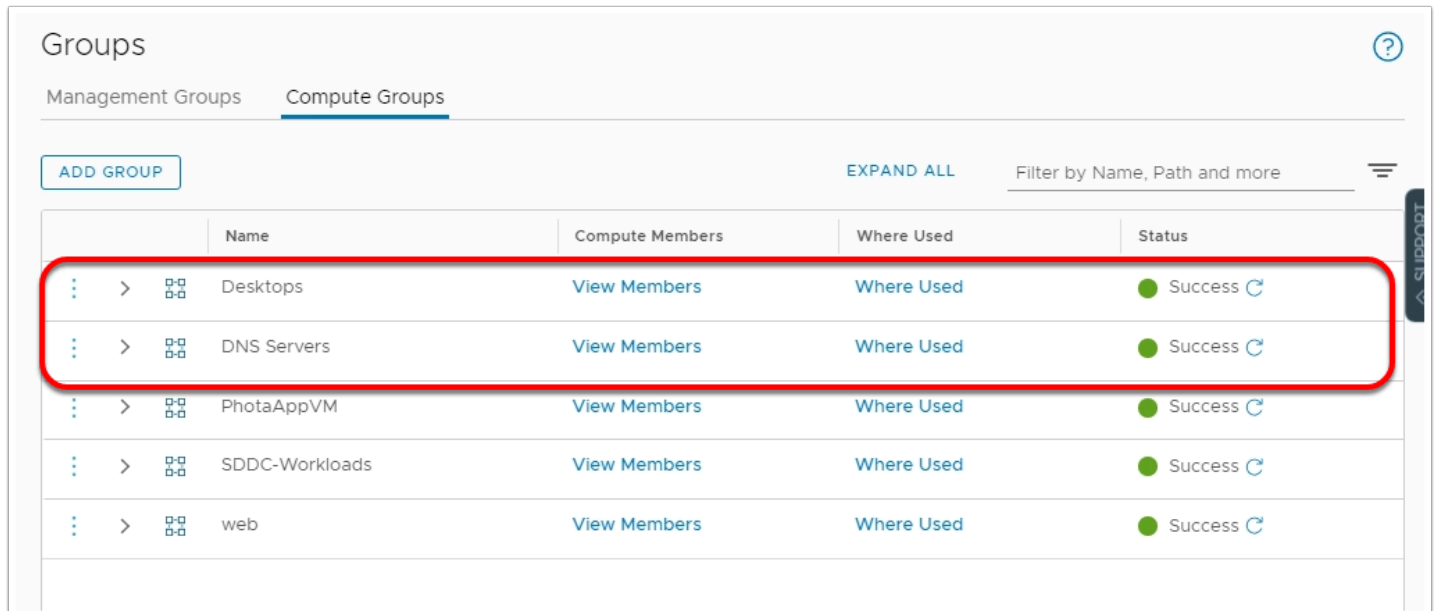
You must set up a DNS rule first, and then the FQDN allowlist or denylist rule below it. This is because NSX-T Data Center uses DNS Snooping to obtain a mapping between the IP address and the FQDN. SpoofGuard should be enabled across the switch on all logical ports to protect against the risk of DNS spoofing attacks. A DNS spoofing attack is when a malicious VM can inject spoofed DNS responses to redirect traffic to malicious endpoints or bypass the firewall













We will start by creating a couple of Groups we will use in the Distributed firewall, followed by a context profile for FQDN filtering and finally, we will define the firewall Policy.

### Task 3.1 - Create Security Groups

1. In the VMC SDDC Console Click the **Networking & Security** Tab
2. Click **Groups**
3. Click **Compute Groups**
4. Click **ADD GROUP**
5. Name the Group **Desktops**
6. Click **Set members**
7. Select the **members** tab
8. Select **Segments** for Category
9. Select **Desktop-Net**
10. Click **APPLY**
11. Click **Save**
12. Using Steps 1 - 11 create a 2nd Group as follows
  - Name: **DNS Servers**
  - Type: **IP Addresses**
    - Values:
      - **192.168.110.10**
      - **8.8.8.8**
      - **8.8.4.4**



Groups				
Management Groups		Compute Groups		
<a href="#">ADD GROUP</a>		<a href="#">EXPAND ALL</a>		<a href="#">Filter by Name, Path and more</a>
	Name	Compute Members	Where Used	Status
	Desktops	<a href="#">View Members</a>	<a href="#">Where Used</a>	<span>Success</span> 
	DNS Servers	<a href="#">View Members</a>	<a href="#">Where Used</a>	<span>Success</span> 
	PhotaAppVM	<a href="#">View Members</a>	<a href="#">Where Used</a>	<span>Success</span> 
	SDDC-Workloads	<a href="#">View Members</a>	<a href="#">Where Used</a>	<span>Success</span> 
	web	<a href="#">View Members</a>	<a href="#">Where Used</a>	<span>Success</span> 

### Task 3.2 - Create FQDN Context Profile & Firewall Policy

1. Under Networking and Security, click **Context Profile**
2. Click **FQDNs** Tab
3. Click **ACTIONS --> Add FQDN**











- Domain: \*.google.com
- Click **SAVE**
  - Click the **Context Profile** Tab
  - Click **ADD CONTEXT PROFILE**
    - Name: **Allowed FQDNs**
    - Click **Set**
      - Click **ADD ATTRIBUTE --> Domain(FQDN) Name**
      - Select the following domains
        - \*.google.com
        - \*.office.com
    - Click **ADD**
  - Click **APPLY**
  - Click **SAVE**

Context Profile

Context Profiles
App IDs
FQDNs

ADD CONTEXT PROFILE

Filter by Name, Path and n

	Name	Attributes	Description	Tags	Where Used
	 360ANTIV	360ANTIV	360 Safeguard is a program developed by Qihoo 360	0	0
	 ACTIVDIR	ACTIVDIR	Microsoft Active Directory	0	0
	 Allowed FQDNs	*.google.com , 2 more		0	0
	 AMQP	AMQP	Advanced Message Queueing Protocol (AMQP) is an application layer protocol which supports business message communication between applications or organizations	0	0
	 AVAST	AVAST	Traffic generated by browsing Avast.com official website of Avast! Antivirus downloads.	0	0

- Click **Distributed Firewall**
- Click **Add Policy**
- Name the Policy **FQDN Whitelist**
- Select the Policy and add 3 firewall rules
- Configure the 3 firewall rules as follows: (Ensure the rules show up in order matching the screenshot)
  - RULE 1
    - Name: **FQDN DNS**
    - Source: **Desktops**
    - Destination: **DNS Servers**
    - Service:

- **DNS**
- **DNS-UDP**

- Context Profile: **DNS**
- Action: **Allow**

## 2. RULE 2

- Name: **Allow limited Public access**
- Source **Desktops**
- Destination: **ANY**
- Service:
  - **HTTP**
  - **HTTPS**
- Context Profile: **Allowed FQDNs**
- Action: **Allow**

## 3. RULE 3

- Name: **Block All HTTP Access**
- Source: **Desktops**
- Destination: **ANY**
- Service:
  - **HTTP**
  - **HTTPS**
- Context Profile: **None**
- Action: **Drop**

## 12. Click **PUBLISH**

Distributed Firewall

All Rules Category Specific Rules

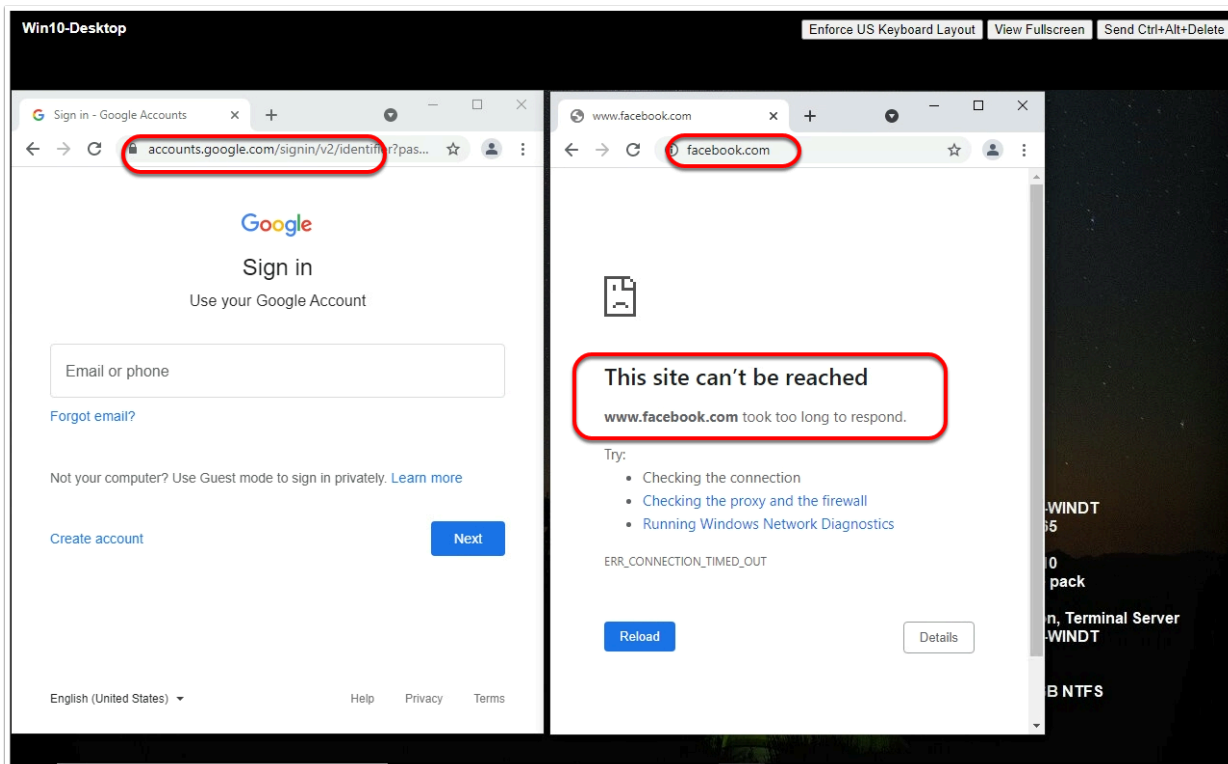
4 Total Unpublished Changes ACTIONS REVERT PUBLISH

ETHERNET (1) EMERGENCY (0) INFRASTRUCTURE (0) ENVIRONMENT (0) APPLICATION (7)

+ ADD POLICY + ADD RULE CLONE UNDO DELETE ... 4 Unpublished Changes Filter by Name, Path and more

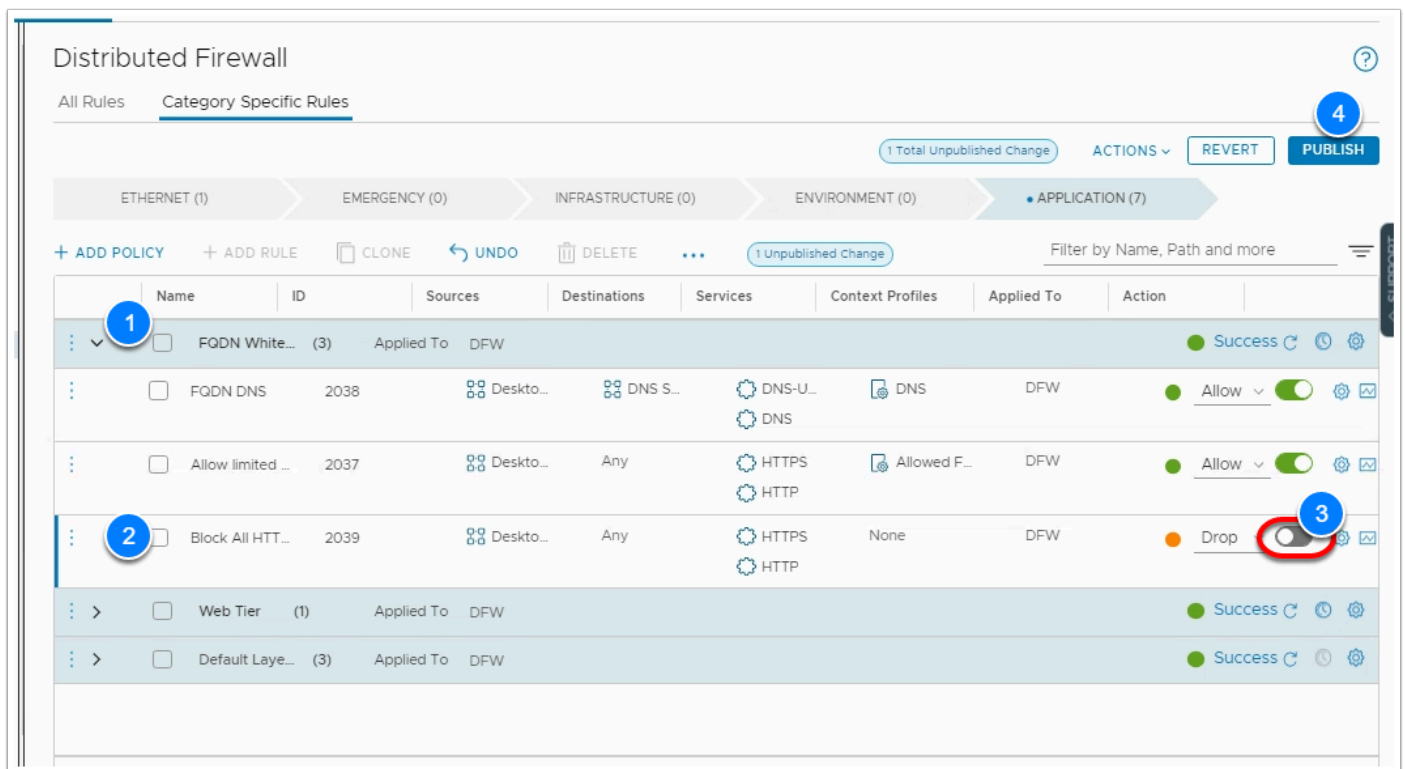
Name	ID	Sources	Destinations	Services	Context Profiles	Applied To	Action
<input checked="" type="checkbox"/> FQDN Whitelist (3)		Applied To	DFW				Success
<input checked="" type="checkbox"/> FQDN DNS	2038	Desktops	DNS Serv...	DNS...	DNS	DFW	Allow
<input checked="" type="checkbox"/> Allow limited Publi...	2037	Desktops	Any	HTT...	Allowed FQDNs	DFW	Allow
<input checked="" type="checkbox"/> Block All HTTP Tra...		Desktops	Any	HTTP	None	DFW	Drop
<input type="checkbox"/> Web Tier (1)		Applied To	DFW				Success
<input type="checkbox"/> Default Layer3 S... (3)		Applied To	DFW				Success

13. In the currently opened browser tab for your SDDC vCenter, login if required (See the setting tab or your worksheet for vCenter credentials)
14. In the inventory select the **Win10-Desktop** VM. (Power it on, if it is Powered-off)
15. Click **LAUNCH WEB CONSOLE**
16. Select the Win10-Desktop Browser tab, Click **Send Ctrl+Alt+Delete** in the upper right-hand of the screen
17. Log into the VM as:
  - **student**
  - **VMware1!**
18. Launch a browser from the Desktop
19. Try accessing **google.com** and **office.com**
20. Try accessing **any other website**



### Task 3.3 - Cleanup

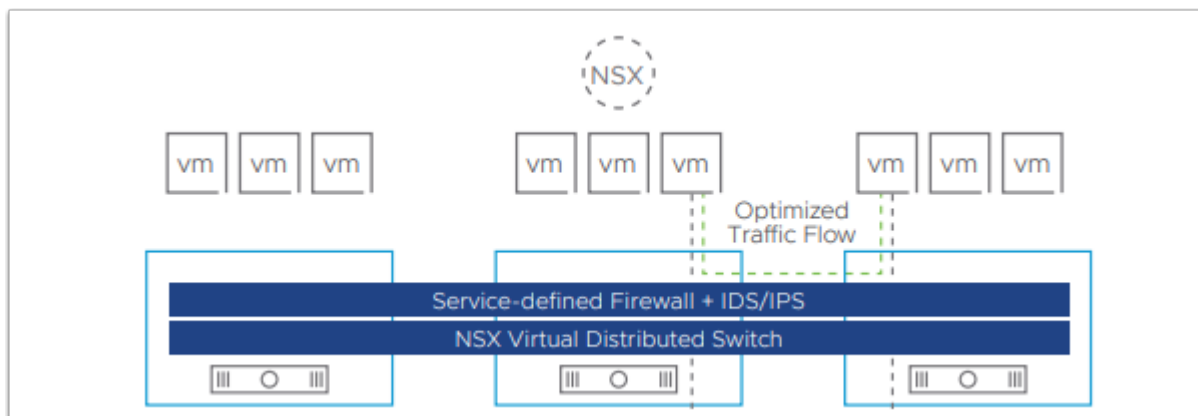
1. Go back to your **SDDC Console** tab
2. Under **Networking and Security**, Click **Distributed Firewall**
3. Expand the **FQDN Whitelist** Policy
4. Select the **Block All HTTP Traffic** rule
5. For the far right of the rule move the **slider** to the left to **disable** it
6. Click **PUBLISH**



## Task 4 - Distributed IDS/IPS

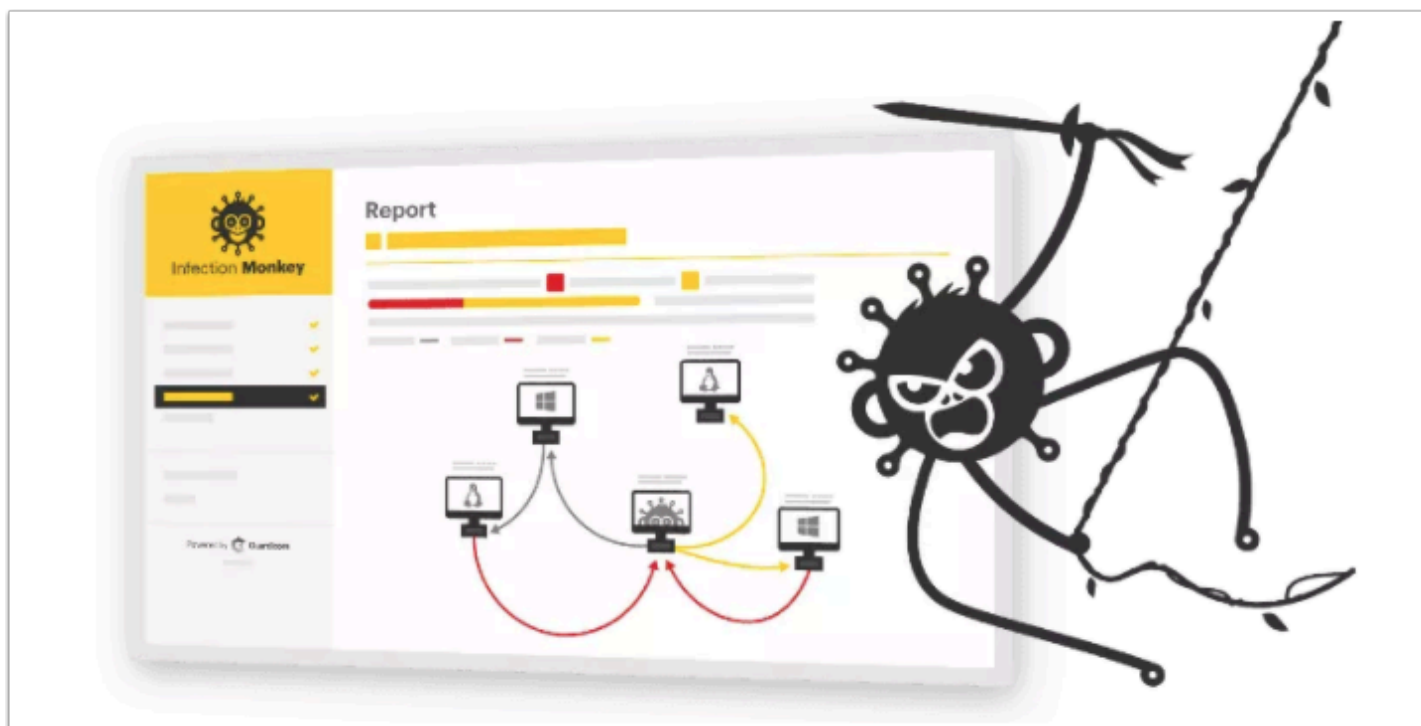
VMware NSX Distributed IDS/IPS provides security operators with a software-based IDS/IPS solution that enables them to achieve regulatory compliance, create virtual zones and detect lateral movement of threats on east-west traffic.

With the rise of distributed applications and micro-services withing VMC on AWS, internal network traffic now dominates traditional north-south traffic. At the same time, the SDDC boundary has diffused with edge and cloud applications as well as with end-user devices. Modern-day attackers noticed these changes and learned to move laterally, aggressively, from their initial point of attack. As a result, inspecting internal east-west (server-to-server) traffic with an advanced threat detection capability is increasingly critical to securing workloads and enterprise data in VMC on AWS.



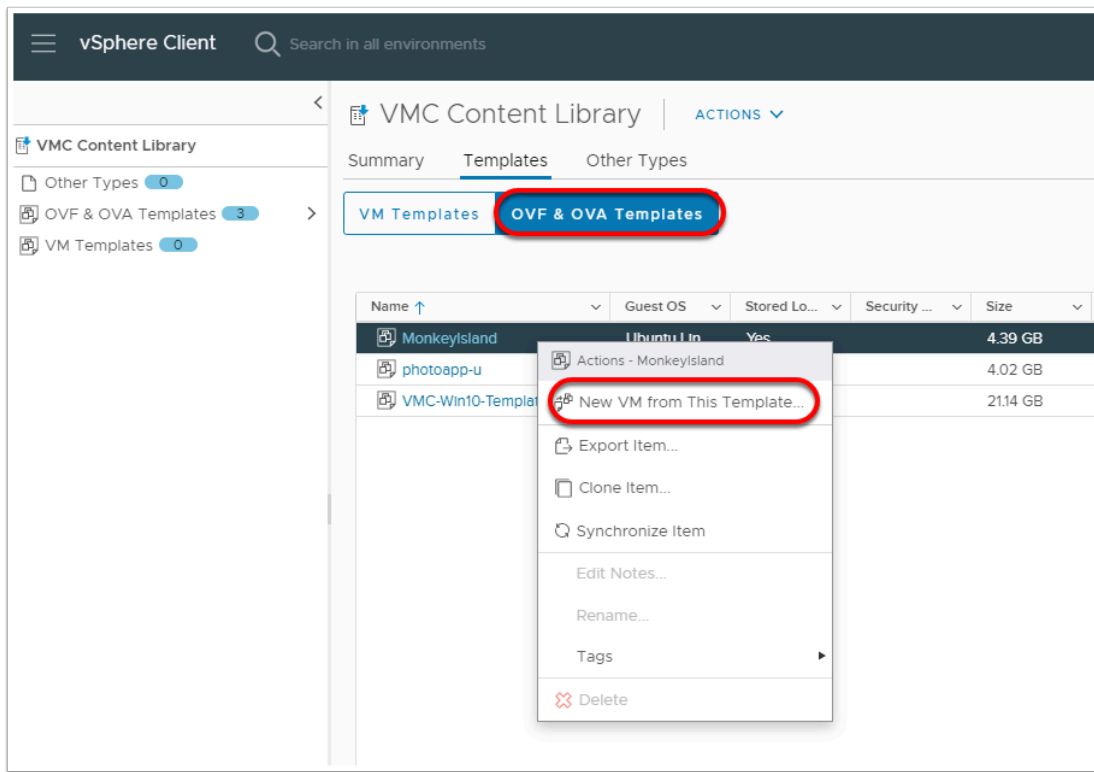
## Task 4.1 - Deploy Infection Monkey virtual Appliance

Infection Monkey is **an open source breach and attack simulation (BAS) platform** that allows organizations to discover security gaps and fix them. You can Simply infect a random machine with the Infection Monkey and automatically discover your security risks. Test for different scenarios - credential theft, compromised machines and other security flaws. We start by deploying Infection Monkey and we'll later use it to exploit a vulnerability.

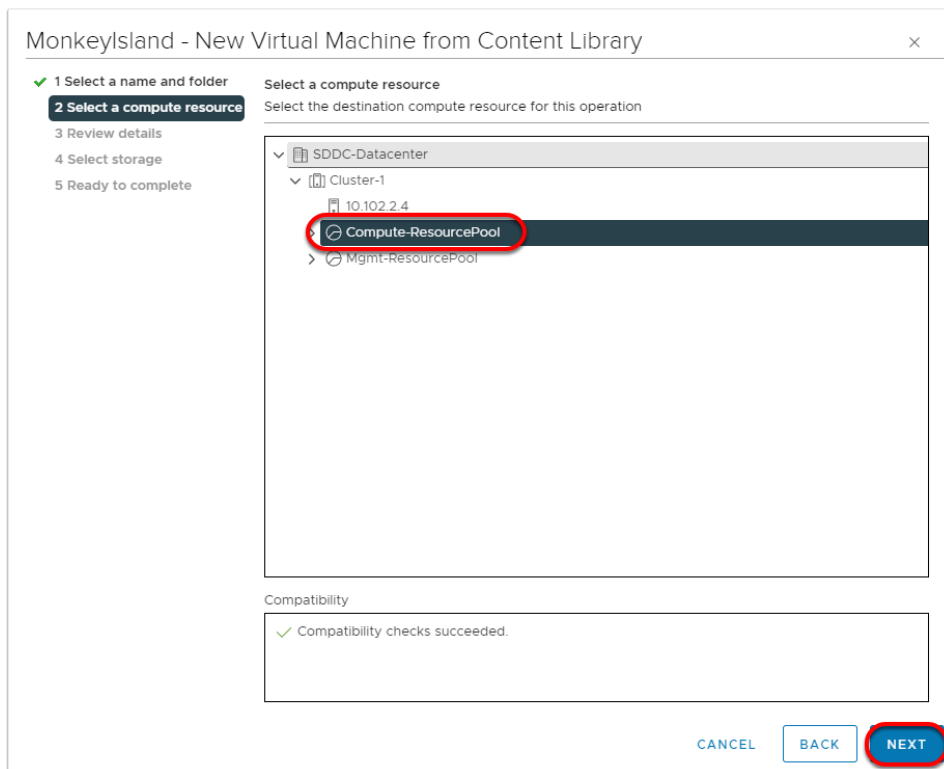


1. From the VDI desktop open your SDDC vCenter browser tab and log in
  - [cloudadmin@vmc.local](#)
  - [<password\\_on\\_the\\_settings\\_tab\\_or\\_your\\_excel\\_worksheet>](#)

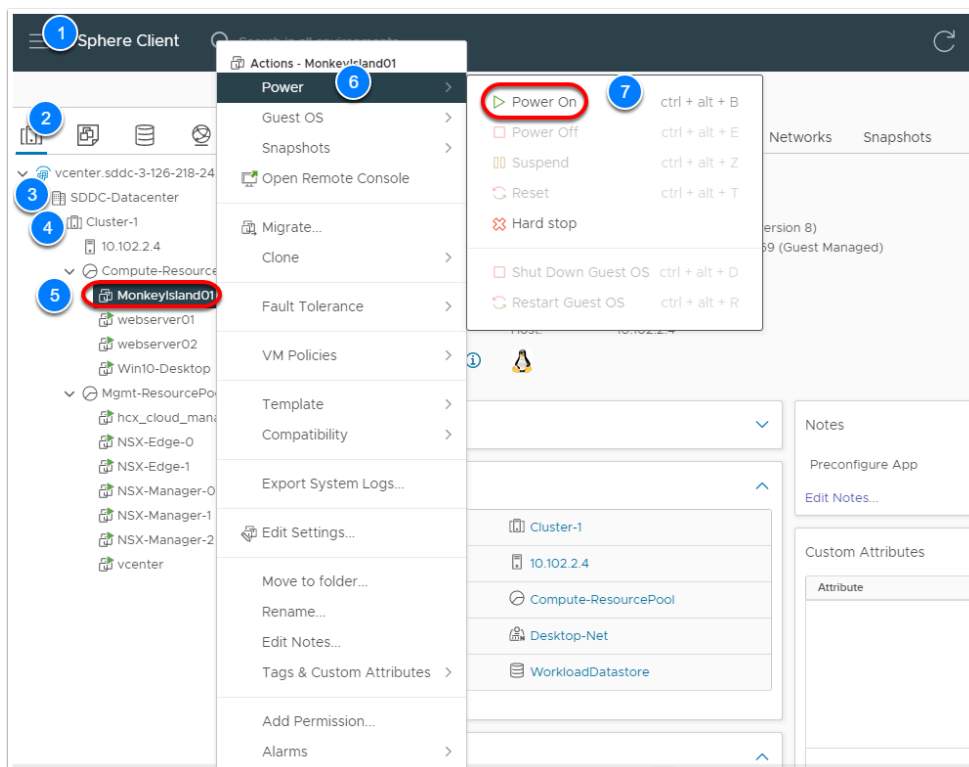
2. Click **Menu --> Content Library**
3. Select the **VMC Content Library**
4. Select the **OVF & OVA Templates** tab
5. right-click **MonkeyIsland**
6. Click **New VM from This Template**



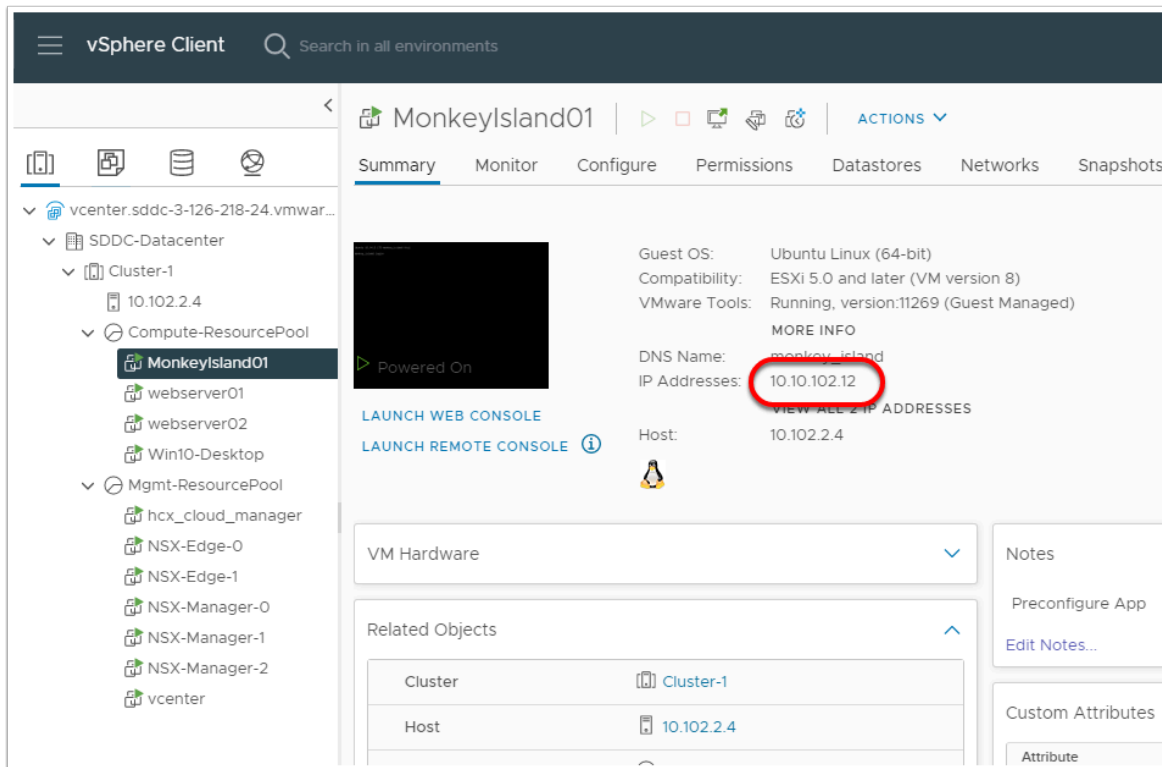
7. Provide the following values in the New Virtual Machine from Content Library Wizard
  - Virtual Machine name: **MonkeyIsland01**
  - Location: **Workloads**
  - Compute resource: **Cluster-1--> Compute-ResourcePool**
  - Storage: **WorkloadDatastore**
  - Network: **Desktop-Net**



8. Monitor the deployment progress, once the MonkeyIsland appliance is deployed, Click the **Hosts & Cluster** view.
9. Expand the **Cluster --> Compute-ResourcePool**
10. Right-click **MonkeyIsland01** choose **Power --> Power-on**

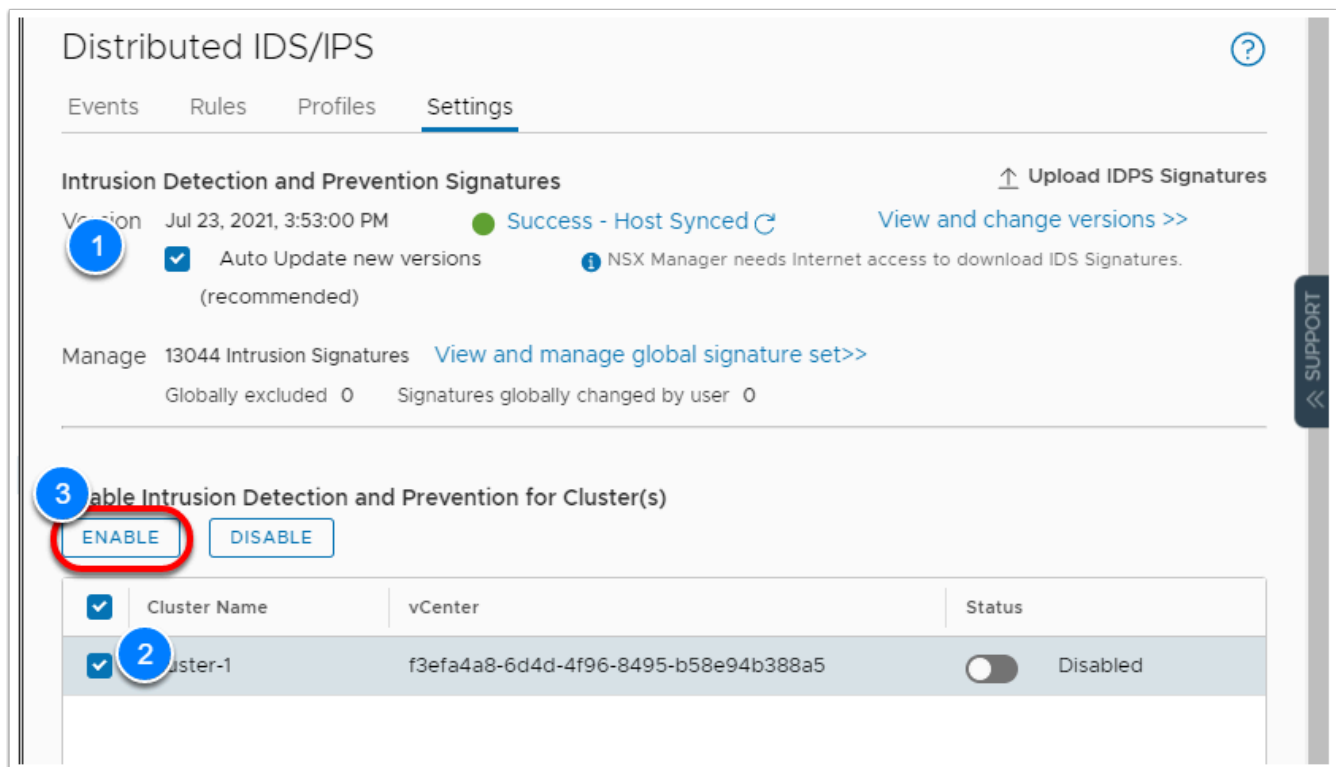
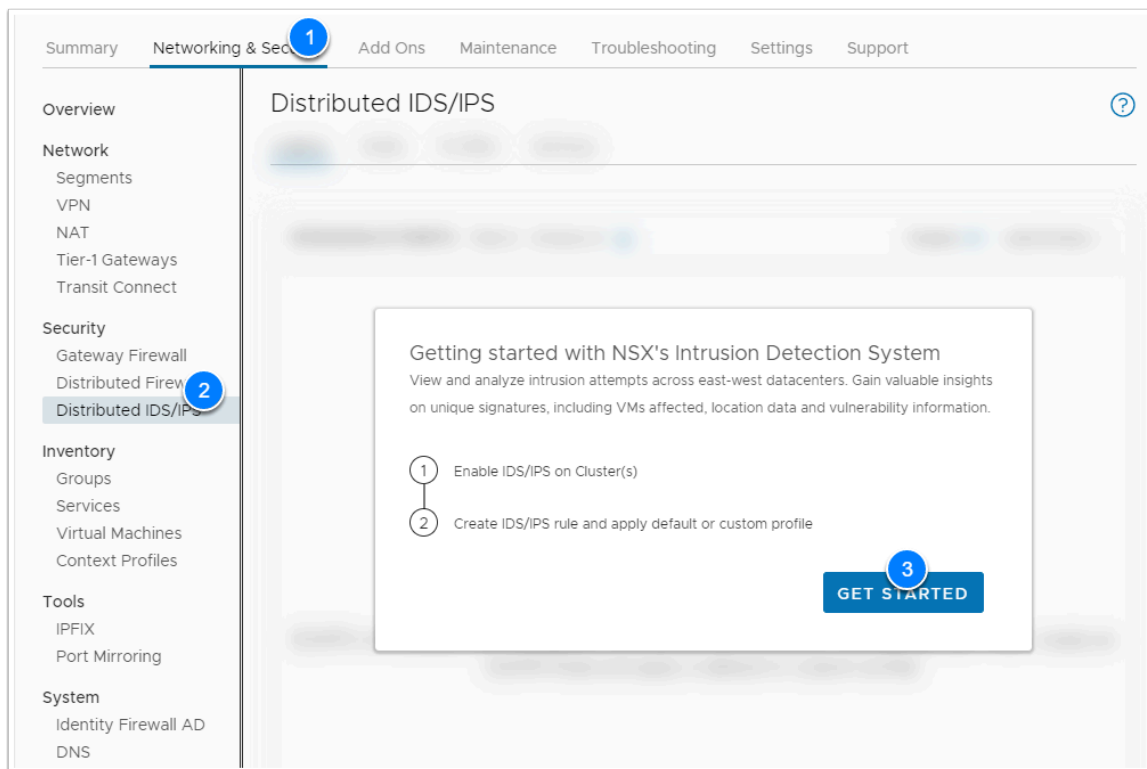


11. Once powered-on, record the **IP Address** of the MonkeyIsland appliance, you will use it later to invoke the exploit.



## Task 4.2 - Enable and Configure Distributed IDS/IPS

1. In the Chrome browser tab for the VMC SDDC Console, Click the **Networking & Security** tab of your SDDC
2. Click **Distributed IDS/IPS**
3. Click **Get Started**
4. Under the Settings tab
  1. Click the **Auto Update versions** checkbox
  2. Check **Cluster-1**
  3. Click **Enable** to enable Distributed IDS/IPS for Cluster-1
5. Click **YES** to confirm you want to Enable D-IDS/IPS



Now, we'll create an IDS/IPS profile to use with an IDS/IPS rule

6. Click the **Profiles** tab under Distributed IDS/IPS
7. Click **ADD PROFILE**
8. Name the Profile **Desktop Profile**
9. Leave the default settings for every other option

10. Click **SAVE**

Distributed IDS/IPS

Events Rules **Profiles** Settings

ADD PROFILE EXPAND ALL Filter by Name, Path and more

Name	Description	Tags	Status
Desktop Profile	Description	Tat Scc	

IDS Signatures Included: 13044 Total: 13044

Intrusion Severities

☒ Critical (4867) ☒ High (8028) ☒ Medium (95) ☒ Low (54)

Additional Options

Filter intrusion signatures to include in this profile by attack type, CVSS and more.

Attack Types Select CVSS Select

Attack Targets Select Products Affected Select

Manage (optional) - change actions and/or exclude signatures specific to this profile. [Manage signatures for this profile >>](#)

Note: the available list of intrusion signatures to customize is based upon the selected attributes above.

SAVE CANCEL

11. Click the **Rules** tab under Distributed IDS/IPS

12. Click **ADD POLICY**

13. Name the Policy **Desktop IDPS Policy**

14. Select the Policy

15. Click **ADD RULE**

16. Set the Rule as follows

- Name: **Desktop Exploit Detection**
- Source: **SDDC-Workloads**
- Destination: **Any**
- Services: **Any**
- IDS Profile: **Desktop Profile**
- Mode: **Detect Only**
- Click the **Gear** at the far right of the rule and **Enable Logging**
- Click **APPLY**

17. Click **PUBLISH**

18. Click the **Events** Tab, we'll return here later to review discovered exploits

Settings

×

Rule > Desktop Exploit Detection

Logging

☒ Enable ⓘ

Direction

In-Out ▾

IP Protocol

☐ IPV4
☐ IPV6
☒ IPV4-IPV6

Log Label

Comments

CANCEL

APPLY

Distributed IDS/IPS

Events

Rules

Profiles

Settings

1 ADD POLICY

2 ADD RULE

CLONE

↶ UNDO

🗑️ DELETE

⋮

2 Total Unpublished Changes

REVERT

5 PUBLISH

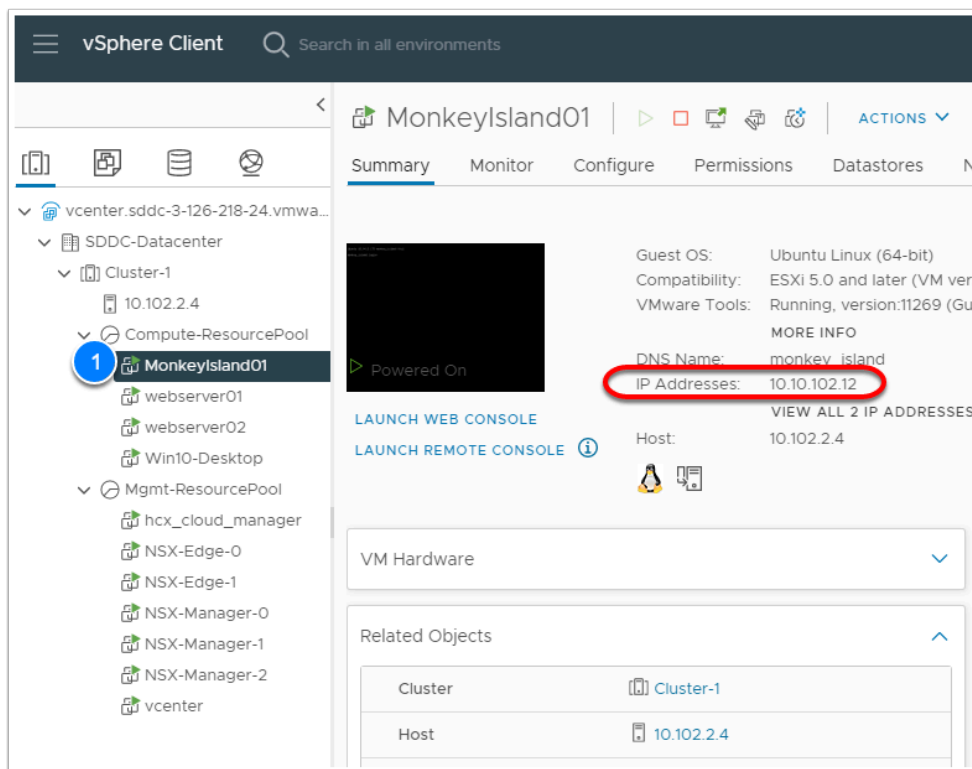
Filter by Name, Path and more

	✓ Name	ID	Sources	Destinations	Services	IDS Profile	Applied To	Mode	
⋮	✓ Desktop IDPS Policy (1)								⚙️
⋮	✓ Desktop Exploit Detection		SDDC-Workloads	Any	Any	Desktop Profile	DFW	Detect Only ▾	<input checked="" type="checkbox"/> <div>4</div>

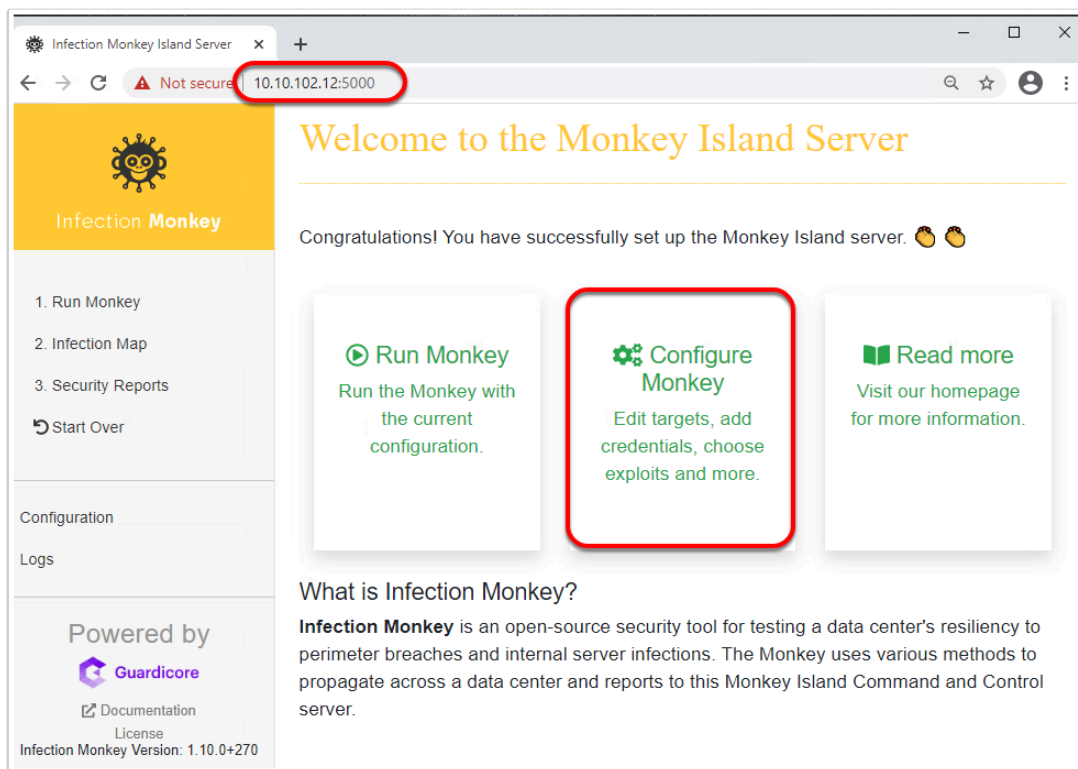
## Task 4.3 - Execute workload Exploits

We will now use Infection Monkey to inject some exploits.

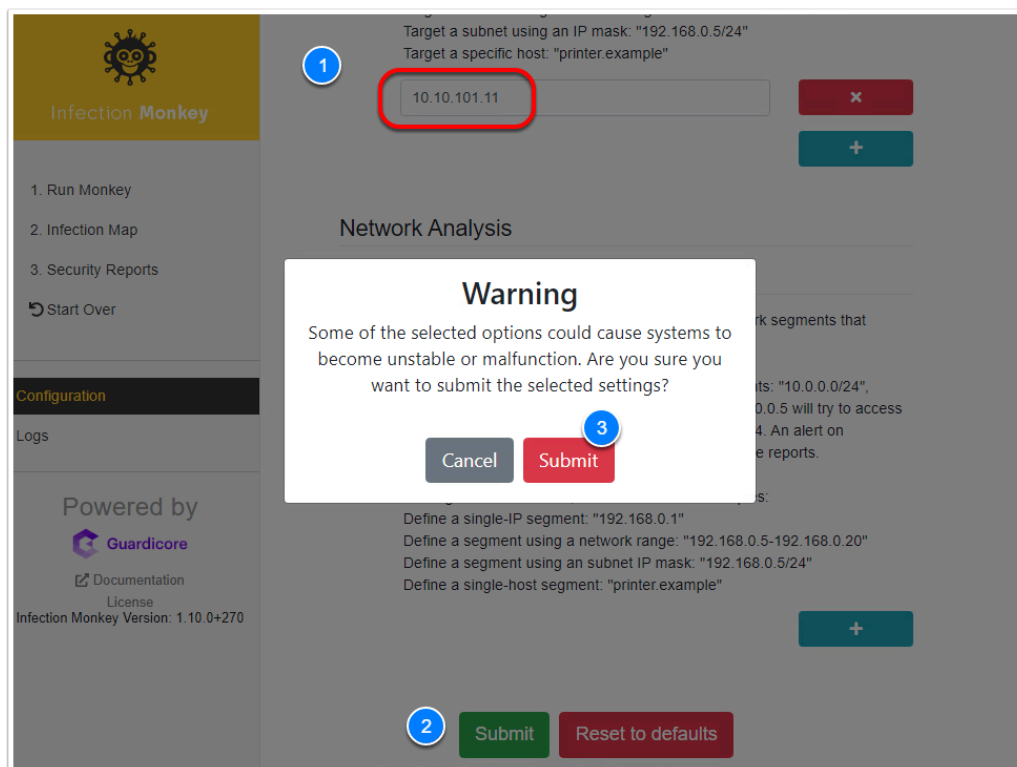
- From the chrome browser tab for the SDDC vCenter, login if required
  - [cloudadmin@vmc.local](mailto:cloudadmin@vmc.local)
  - [<password\\_on\\_the\\_settings\\_tab\\_or\\_your\\_excel\\_worksheet>](#)
- Select the **MonkeyIsland01** VM
- Confirm the VM is powered-on and take note of the **IP address** if you hadn't done so previously
- Select **Win10-Desktop**
- Take note of the **IP Address**
- Click **LAUNCH WEB CONSOLE**



5. In the browser tab for Win10-Desktop Console, login as:
  - **student**
  - **VMware1!**
6. Launch **Chrome** from the **Win10-Desktop**
7. In the address bar type, [https://<MonkeyIsland\\_IPAddress>:5000](https://<MonkeyIsland_IPAddress>:5000)
8. login as:
  - **monkeyuser**
  - **VMware1!**
9. Click **Configure Monkey**

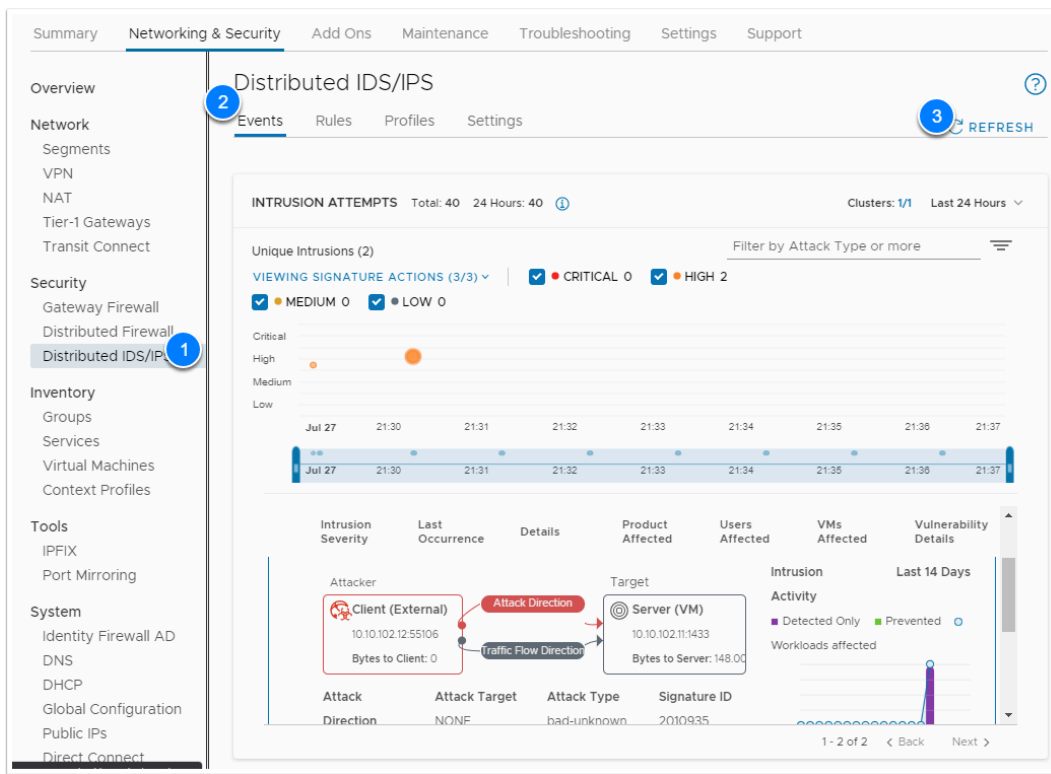


10. Click the **Network** tab
11. Change the Scan Target list IP address to the <IP address of your Win10-Desktop VM>
12. Click **Submit**
13. Click **Submit** Again, to save your change



14. In the left pane, click **Run Monkey**

15. Click **From Island**
16. Click **Infection Map**, to watch Infection Monkey run its exploits.  
Note: the exploits take about 10 mins to run
17. Click the VDI browser tab for the SDDC Console
18. On the **Events** tab of the Distributed IDS/IPS Click **refresh** to review the Intrusion attempts



**NOTE:** When setting up IDS/IPS policies you 1st want to use the “Detect only” action to identify all possible exploits and the possible vulnerable systems. Once you have a good handle on what could be exploited you can then move to tune the Policies and rules and as part of that start to use the “Detect and Prevent” action. In essence, starting with “Detect and Prevent” action could have unwanted and even disastrous effects on a production environment without 1st exploring the necessary due diligence.

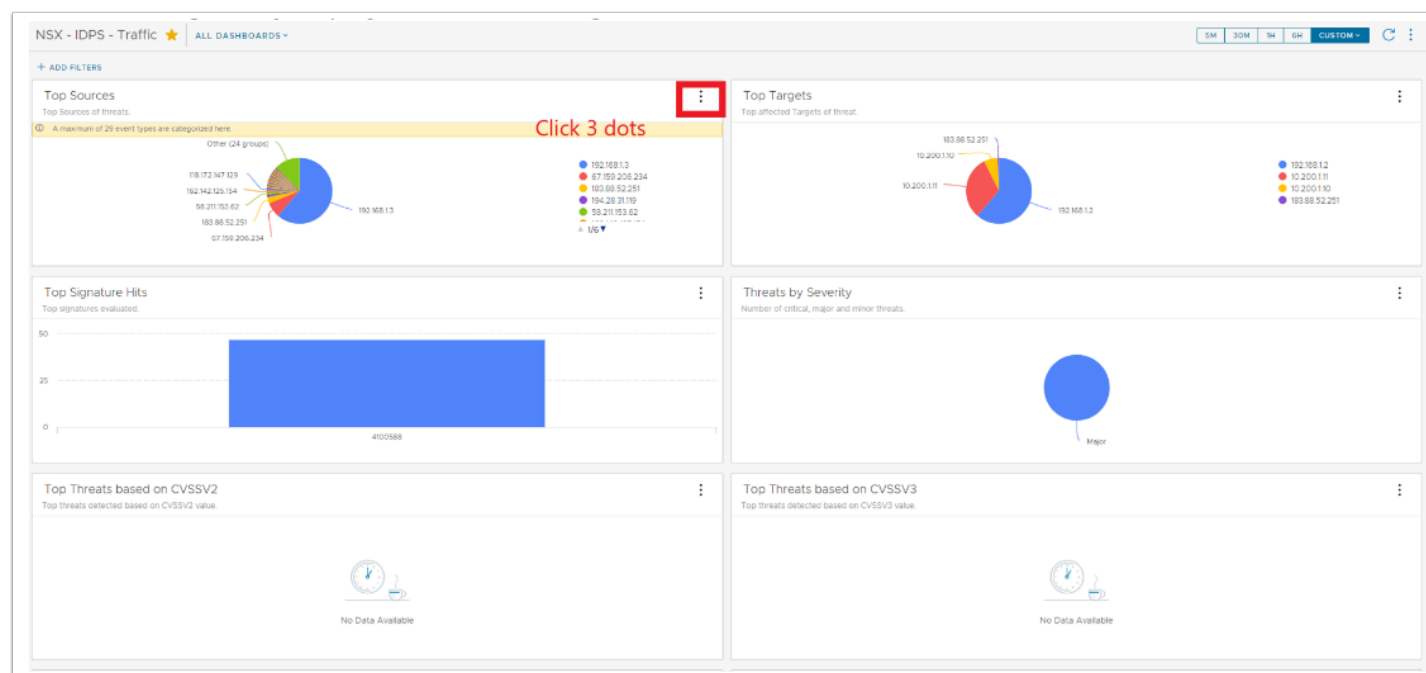
Many customers tend to implement only a handful of "Detect and Prevent" rules and typically they would be tied to their critical systems for known exploits, while having Detect only rule for most things. This allows them to better identify and address false-positive that could impede production activities.

## Task 4.4 - Review Intrusion Event In Log Insight

vRealize log Insight Cloud is the one platform capable of bringing log data from your entire environment together, no matter where it resides, and extracting meaning from it. The solution brings order to the chaos of millions of unstructured data points, turning raw log data into actionable insights that can help you address both security and operational issues.

Log Insight Cloud contains built-in knowledge and native support for VMware SDDC technologies—from VMware vSphere, NSX-T firewall, VMware Cloud on AWS logs, AWS Services, to 3rd party technologies as Docker, MS SQL, Apache and many more.

Log Insight Cloud can also capture and Analyze VMC on AWS Intrusion events detected by the NSX advances security Add-on. All you need to do is enable Logging in your Distributed IDS/IPS rule(s). In Task 4.2 you enabled logging for your IDS/IPS rule, now we will review the log entry in Log Insight.



1. In the SDDC Console, click **Activity Logs**
2. Click the **vRealize Log Insight Cloud** Link

## Activity Log

You can view additional events and log data in [vRealize Log Insight Cloud](#)

Event Name	Event Type	Time	SDDC Name
DNS Update of Management IP	Activity	11/11/21, 12:47 PM	VMC_Ninja
SDDC Group Deletion	Activity	11/10/21, 3:42 PM	
SDDC Member Deleted from Group	Activity	11/10/21, 3:32 PM	
Updating Management VM	Activity	11/10/21, 12:06 PM	VMC_Ninja
Updating Management VM	Activity	11/10/21, 12:01 PM	VMC_Ninja
Updating Management VM	Activity	11/10/21, 11:58 AM	VMC_Ninja
Updating Management VM	Activity	11/10/21, 11:52 AM	VMC_Ninja

3. In the Left Pane Click **Dashboard**
4. Click **NSX - IDPS Traffic** to view the dashboard. This will display the top sources, destinations, signatures, etc...
5. In the upper-right, click **6H** to adjust the time scale. The Dashboard defaults to the past 5 minutes.

## NSX - IDPS Traffic

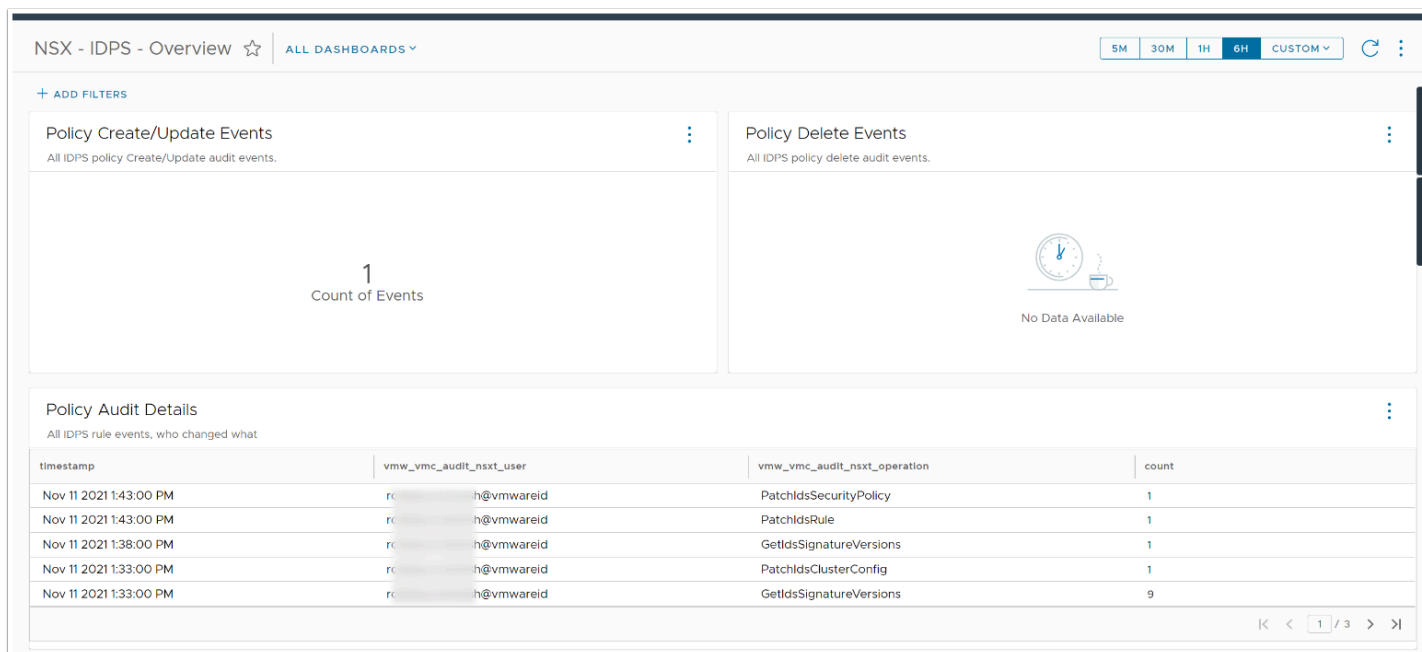
Top Sources of threats.

Top affected Targets of threat.

Top Signature Hits

Threats by Severity

6. Repeat the steps above to view the **NSX - IDPS - Overview** Dashboard



## Conclusion

The NSX Advanced Distributed Security for VMware Cloud on AWS workloads ensure workloads are secure and compliance goals are met. NSX Advanced Firewall for VMware Cloud on AWS customers provides layer 7 distributed security that scales linearly with VMs, with no blind spots during network traffic inspections. With The NSX Advanced Firewall enabled, you can make use of:

- **Distributed Firewall with Layer 7 Application ID** - Deep Packet Inspection built into the hypervisor with built in profiles for common enterprise applications.
- **Distributed Firewall with Active Directory based User ID** - Per user and session application access control with an Identity Firewall
- **Distributed Firewall with FQDN Filtering** - Permit or deny communication to specific destinations in the Internet.
- **Distributed Firewall with Active Directory based User ID** - Per user and session application access control with an Identity Firewall.