

Configuring a Network with SIEM Detection and Monitoring

Description

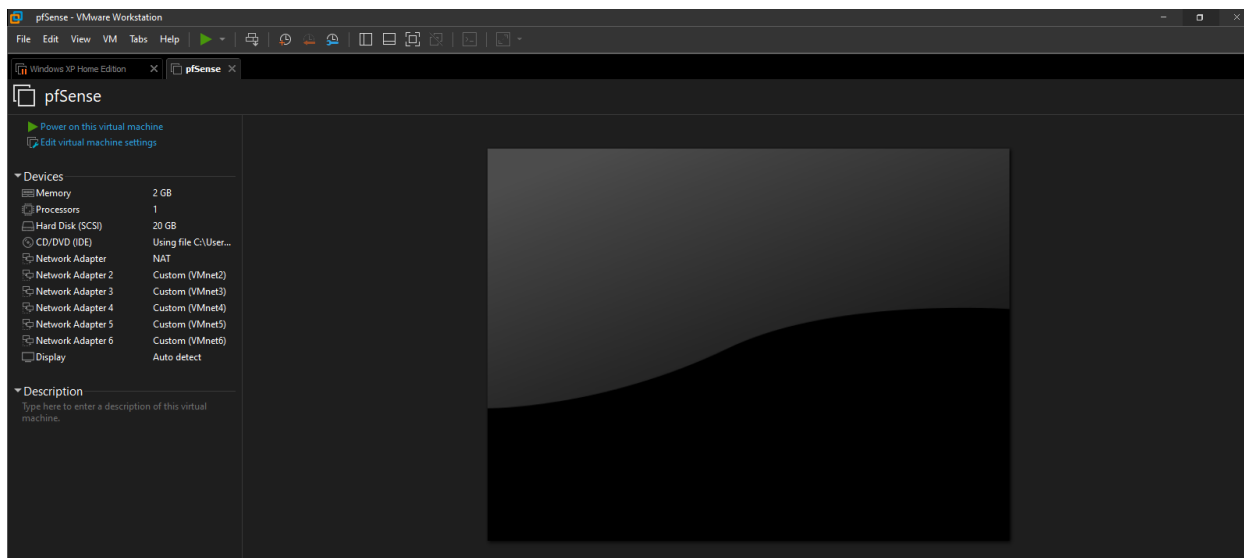
The main purpose of this lab is to gain hands-on experience with network configuration along with detecting and monitoring attacks within a controlled, virtual network by monitoring alerts and reading logs through a SIEM. The lab involves configuring a small network consisting of a victim network with one user and a Domain Controller, two SIEM tools connected separately to the victim network for security monitoring and log management of said network, a Kali Linux machine as the attacker machine, and a network firewall that will connect to all of the other virtual networks and acts as a default gateway, which will filter and segment the network.

Programs and Environments Used

- Hypervisor: VMWare Workstation 16 Pro
- Firewall: pfSense
- SIEM Tools: Security Onion, Splunk
- Servers: Windows Server 2019 (Domain Controller), Ubuntu Server
- Desktops: Windows 10 Enterprise, Ubuntu Desktop

pfSense Configuration

pfSense will be configured as a network firewall in order to segment the network, which can only be accessed from our Kali Linux machine for additional configuration.



For this network, pfSense will require 6 network adapters in total to support the other machines/virtual networks created in the later portion of the lab as seen above.

```
Starting syslog...done.
Starting CRON... done.
pfSense 2.6.0-RELEASE amd64 Mon Jan 31 19:57:53 UTC 2022
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

VMware Virtual Machine - Netgate Device ID: 3988b6f0cc5320bc96e2

*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> em0      -> v4/DHCP4: 192.168.232.129/24
LAN (lan)      -> em1      -> v4: 192.168.1.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults    13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: █
```

After accepting all of the default settings for the pfSense installation, pfSense will load this screen above. The LAN address will be the IP address for pfSense once we configure it. From here, we will assign the pfSense interfaces to the corresponding network interfaces for the other virtual networks as seen below:

```
NOTE: this enables full Firewalling/NAT mode.
(em1 em2 em3 em4 em5 a or nothing if finished): em1

Enter the Optional 1 interface name or 'a' for auto-detection
(em2 em3 em4 em5 a or nothing if finished): em2

Enter the Optional 2 interface name or 'a' for auto-detection
(em3 em4 em5 a or nothing if finished): em3

Enter the Optional 3 interface name or 'a' for auto-detection
(em4 em5 a or nothing if finished): em4

Enter the Optional 4 interface name or 'a' for auto-detection
(em5 a or nothing if finished): em5

The interfaces will be assigned as follows:

WAN -> em0
LAN -> em1
OPT1 -> em2
OPT2 -> em3
OPT3 -> em4
OPT4 -> em5

Do you want to proceed [y|n]? █
```

The listed pfSense interfaces will correspond with the following virtual network:

WAN (NAT) → WAN

LAN (Vmnet2) → Kali

OPT1 (Vmnet3) → Victim Network

OPT2 (Vmnet4) → Security Onion

OPT3 (Vmnet5) → Span Port

OPT4 (Vmnet6) → Splunk

After assigned the pfSense interfaces to the corresponding virtual networks, pfSense will load this screen below, which shows we need to assign IP addresses to the remaining interfaces:

```
Writing configuration...done.
One moment while the settings are reloading... done!
VMware Virtual Machine - Netgate Device ID: 3988b6f0cc5320bc96e2

*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> em0      -> v4/DHCP4: 192.168.232.129/24
LAN (lan)      -> em1      -> v4: 192.168.1.1/24
OPT1 (opt1)    -> em2      ->
OPT2 (opt2)    -> em3      ->
OPT3 (opt3)    -> em4      ->
OPT4 (opt4)    -> em5      ->

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM
8) Shell

Enter an option: █
```

To do so, we will enable DHCP server on the LAN network, and set a range of IP addresses for assignment to the other interfaces as shown below:

```

Enter the new LAN IPv4 address. Press <ENTER> for none:
> 192.168.1.1

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.
e.g. 255.255.255.0 = 24
      255.255.0.0   = 16
      255.0.0.0     = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
>

Enter the new LAN IPv6 address. Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? (y/n) y
Enter the start address of the IPv4 client address range: 192.168.1.11
Enter the end address of the IPv4 client address range: 192.168.1.200
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) █

```

After finish configuring the DHCP server to the LAN network, pfSense can now be accessed to its assigned IP address 192.168.1.1 through a web browser, which we will see in the later part of the lab via Kali Linux. The finished setup can be seen below:

```

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
>

Enter the new LAN IPv6 address. Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? (y/n) y
Enter the start address of the IPv4 client address range: 192.168.1.11
Enter the end address of the IPv4 client address range: 192.168.1.200
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) n

Please wait while the changes are saved to LAN...
Reloading filter...
Reloading routing configuration...
DHCPD...

The IPv4 LAN address has been set to 192.168.1.1/24
You can now access the webConfigurator by opening the following URL in your web
browser:
      https://192.168.1.1/

Press <ENTER> to continue. █

```

After configuring a DHCP pool of available IP addresses used for assignment, we will continue assigning IP addresses to the remaining interfaces on pfSense. OPT3 (Span Port) will be left unassigned as it is going to have Span Port with traffic that is monitored from the victim network by Security Onion. Therefore, the final pfSense configuration can be seen below:

```

The IPv4 OPT4 address has been set to 192.168.4.1/24

Press <ENTER> to continue.
VMware Virtual Machine - Netgate Device ID: 3988b6f0cc5320bc96e2

*** Welcome to pfSense 2.6.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> em0      -> v4/DHCP4: 192.168.232.129/24
LAN (lan)      -> em1      -> v4: 192.168.1.1/24
OPT1 (opt1)    -> em2      -> v4: 192.168.2.1/24
OPT2 (opt2)    -> em3      -> v4: 192.168.3.1/24
OPT3 (opt3)    -> em4      ->
OPT4 (opt4)    -> em5      -> v4: 192.168.4.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: █

```

Security Onion Configuration

For this section, I was not able to complete it after waiting for Security Onion to finish its installation due to how much memory it requires, and my host machine is not capable of allocating a high amount of RAM for both Security Onion and an Ubuntu desktop virtual machine running at the same time (Ubuntu desktop is used to access the SIEM web interface of Security Onion once its installation is complete). Instead, I had a friend who also worked on this lab with me and provided some screenshots of the Security Onion configuration.

Once the installation of Security Onion is fully complete, run the “sudo so-allow” command and select option [a] once this menu shows up as seen below:

```

This program allows you to add a firewall rule to allow connections from a new I
P address.

Choose the role for the IP or Range you would like to add

[a] - Analyst - ports 80/tcp and 443/tcp
[b] - Logstash Beat - port 5044/tcp
[c] - Elasticsearch REST API - port 9200/tcp
[f] - Strelka frontend - port 57314/tcp
[o] - Osquery endpoint - port 8090/tcp
[s] - Syslog device - 514/tcp/udp
[w] - Wazuh agent - port 1514/tcp/udp
[p] - Wazuh API - port 55000/tcp
[r] - Wazuh registration service - 1515/tcp

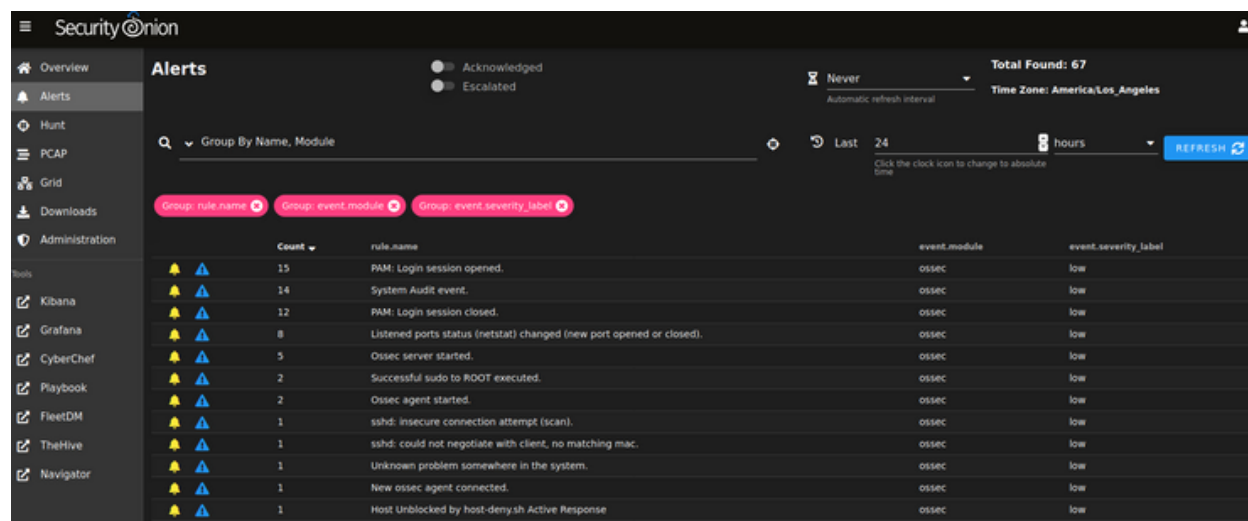
Please enter your selection:

```

 a_

From here, type in the IP address associated with the Ubuntu Desktop. This will create a firewall rule on Security Onion that will allow the Ubuntu Desktop access to the SIEM web interface straight from the desktop.

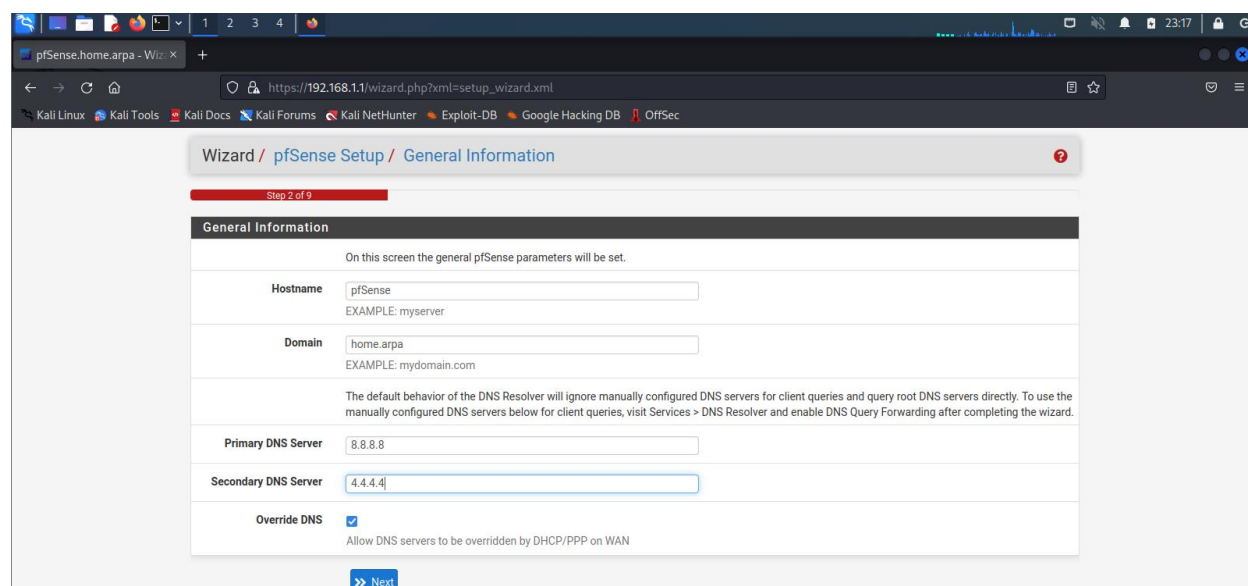
Afterwards, navigate to the Security Onion's IP address on the Ubuntu Desktop's web browser, and you will have access to the SIEM web interface of Security Onion as seen below:



Count	rule.name	event.module	event.severity_label
15	PAM: Login session opened.	ossec	low
14	System Audit event.	ossec	low
12	PAM: Login session closed.	ossec	low
8	Listened ports status (netstat) changed (new port opened or closed).	ossec	low
5	Ossec server started.	ossec	low
2	Successful sudo to ROOT executed.	ossec	low
2	Ossec agent started.	ossec	low
1	sshd: insecure connection attempt (scan).	ossec	low
1	sshd: could not negotiate with client, no matching mac.	ossec	low
1	Unknown problem somewhere in the system.	ossec	low
1	New ossec agent connected.	ossec	low
1	Host Unblocked by host-deny.sh Active Response	ossec	low

Configuring pfSense Interfaces and Rules Through Kali Linux

With pfSense configured earlier, one can access the pfSense WebConfigurator through a web browser in order to make changes to pfSense interface and firewall rules. To do so, we will search for pfSense's IP address (192.168.1.1) in Firefox. After login into pfSense, we will receive this webpage for initial setup as seen below:



Wizard / pfSense Setup / General Information

Step 2 of 9

General Information

On this screen the general pfSense parameters will be set.

Hostname: pfSense
EXAMPLE: myserver

Domain: home.arpa
EXAMPLE: mydomain.com

The default behavior of the DNS Resolver will ignore manually configured DNS servers for client queries and query root DNS servers directly. To use the manually configured DNS servers below for client queries, visit Services > DNS Resolver and enable DNS Query Forwarding after completing the wizard.

Primary DNS Server: 8.8.8.8

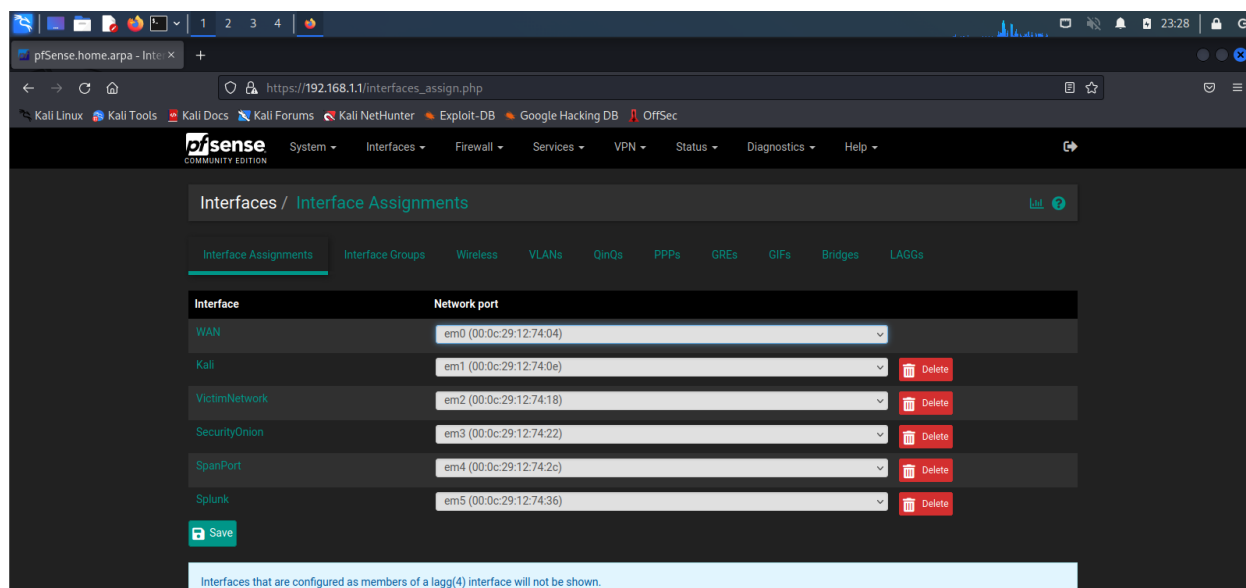
Secondary DNS Server: 4.4.4.4

Override DNS: ☒
Allow DNS servers to be overridden by DHCP/PPP on WAN

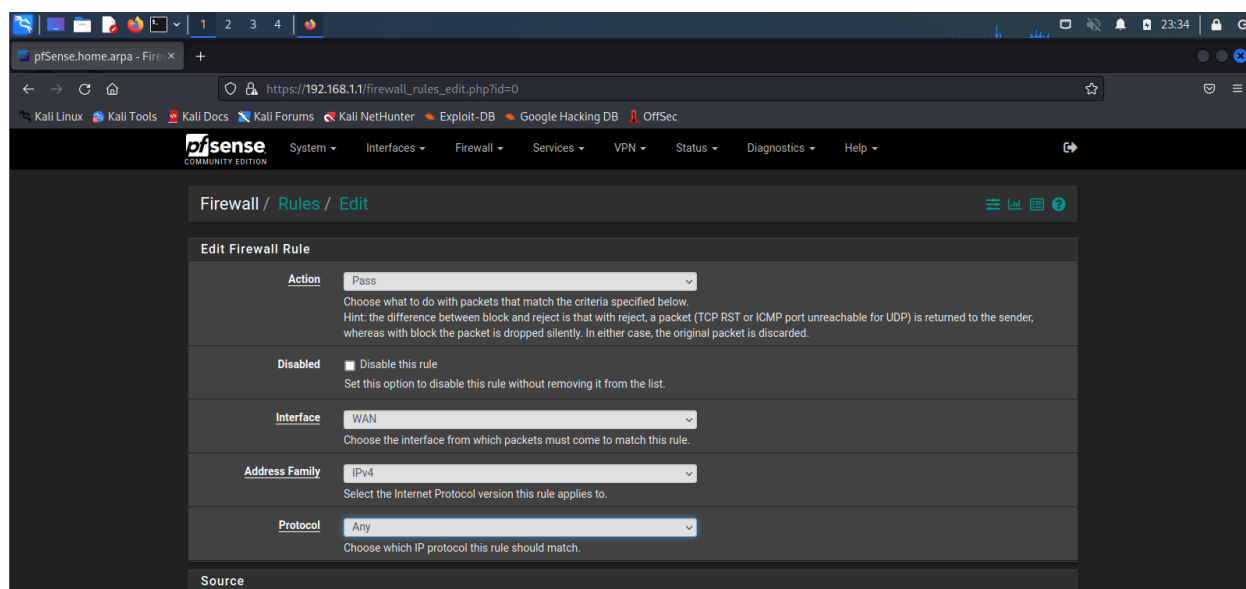
>> Next

From here, we will assign pfSense's primary and secondary DNS servers as shown above. 8.8.8.8 and 4.4.4.4 are Google's DNS servers.

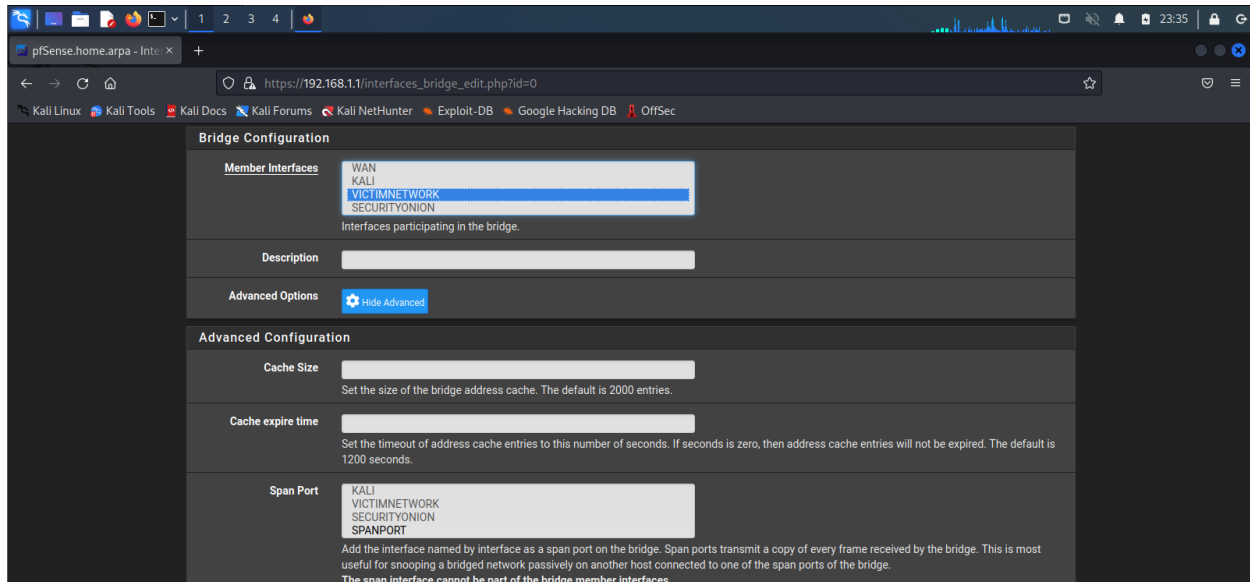
Afterwards, once we finish the initial setup, we will go to interfaces assignments to view all of our assigned interfaces from earlier. From here, we will change the pfSense default interface names to the ones that corresponds to their virtual network in our lab (listed previously in pfSense Configuration) as seen below:



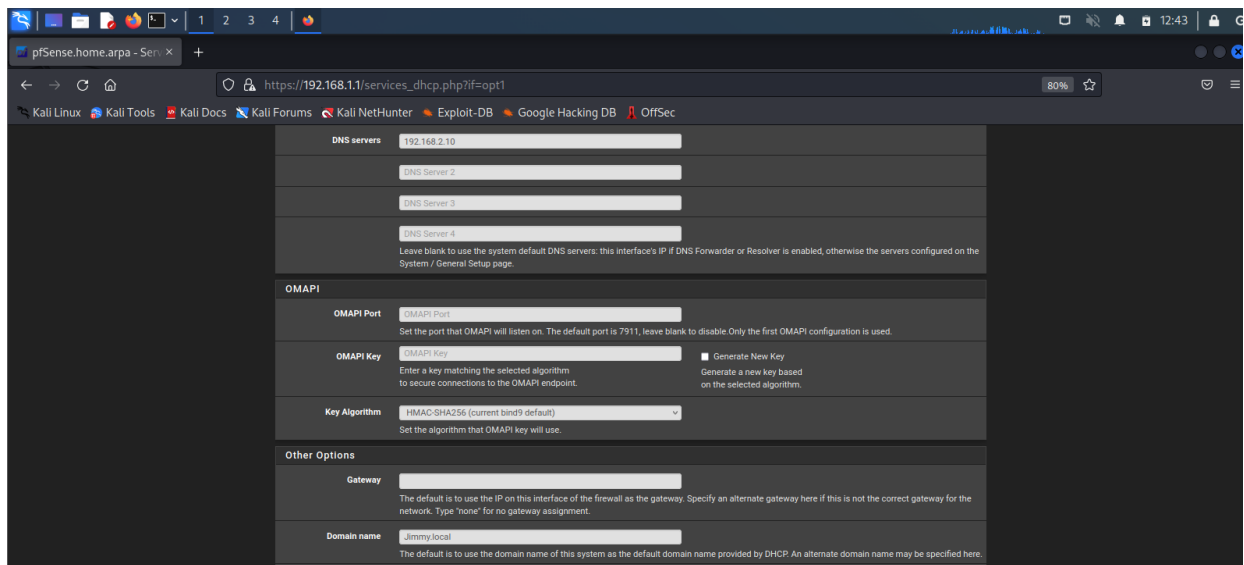
Next, we will create a firewall rule where the pfSense will accept any protocols pass through the firewall as shown below. This will allow for interesting types of security alerts and logs to view when we conduct an attack and monitor/detect it:



We will also want to create a bridge where the member interface is the victim network with the Span Port set as shown below:



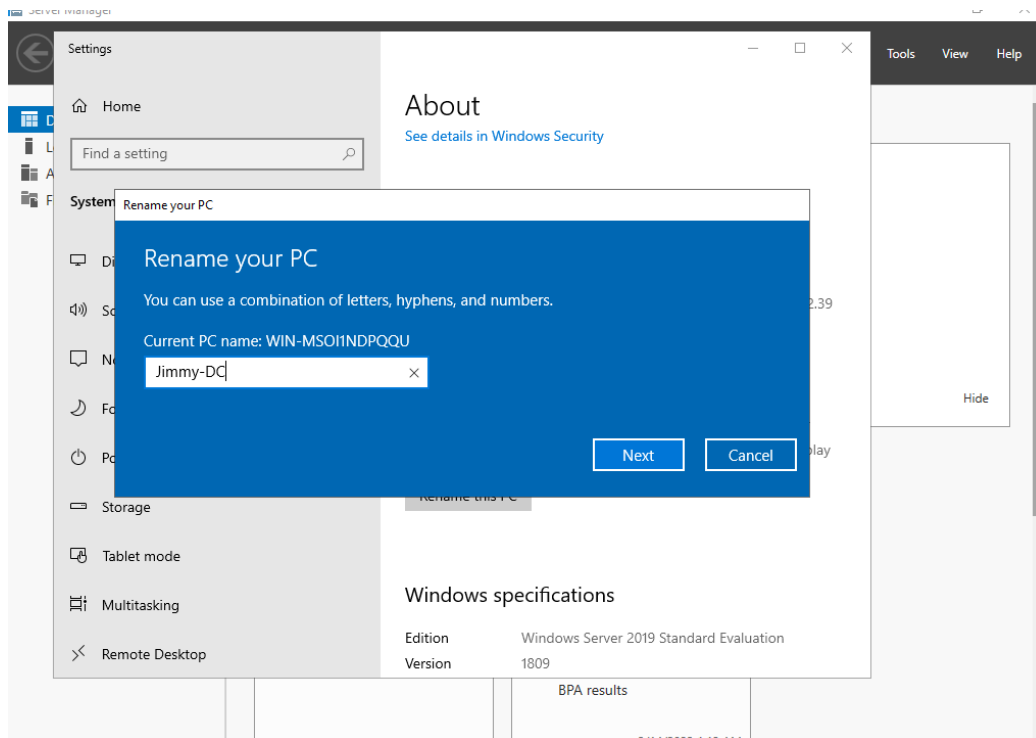
Lastly, we will need to add the DNS server to the DHCP server for the victim network. In addition, we will also add the domain name “Jimmy.local” as shown below. This DNS server IP address and domain name will be from the Domain Controller that will be set up in the next section of this lab. These settings are necessary in order for the Windows User to join the local Active Directory domain in the later section of the lab:



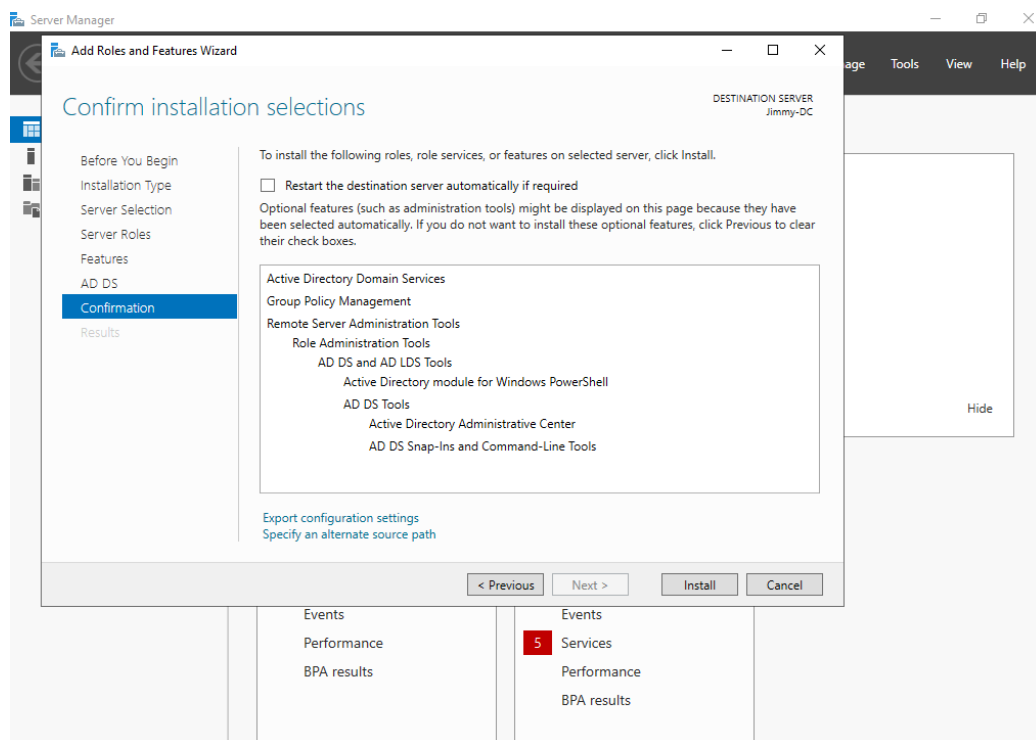
Windows Server Domain Controller Configuration

In this section, we will be setting an local Active Directory domain with a Windows 2019 Server as the Domain Controller with one Windows 10 user. Once we are finished with configuring the initial setup of making an Administrator account, we will change the PC

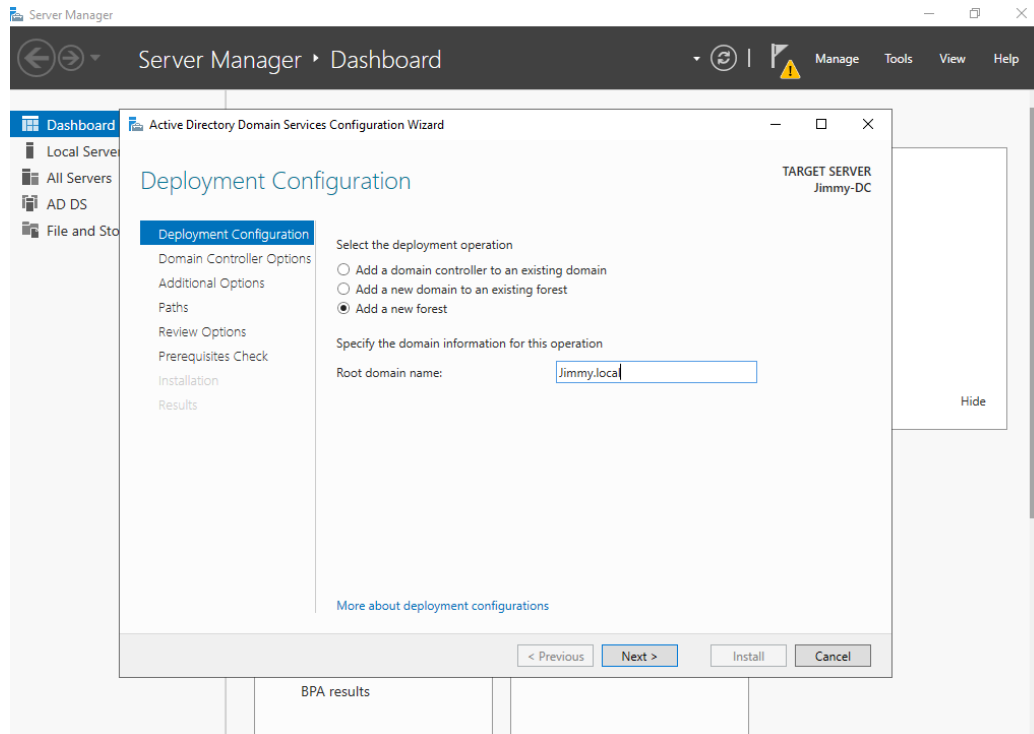
name of the Windows Server to something that we can resemble it is a Domain Controller as shown below:



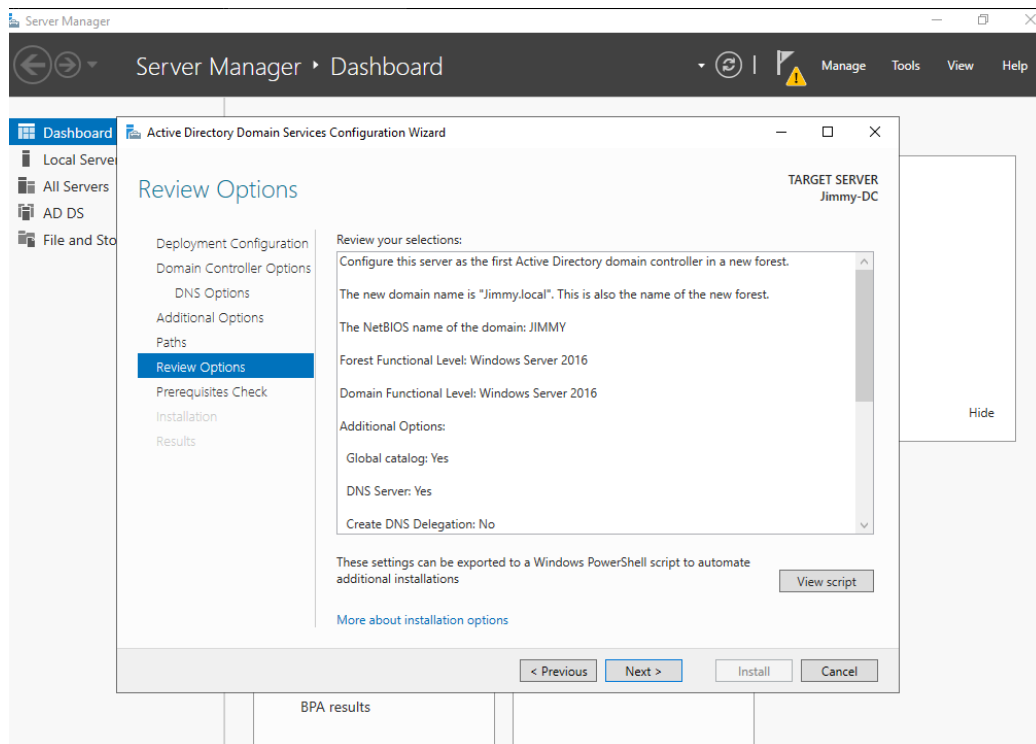
Next, we will want to add the Active Directory Domain Services (AD DS) server role with the default features as shown below:



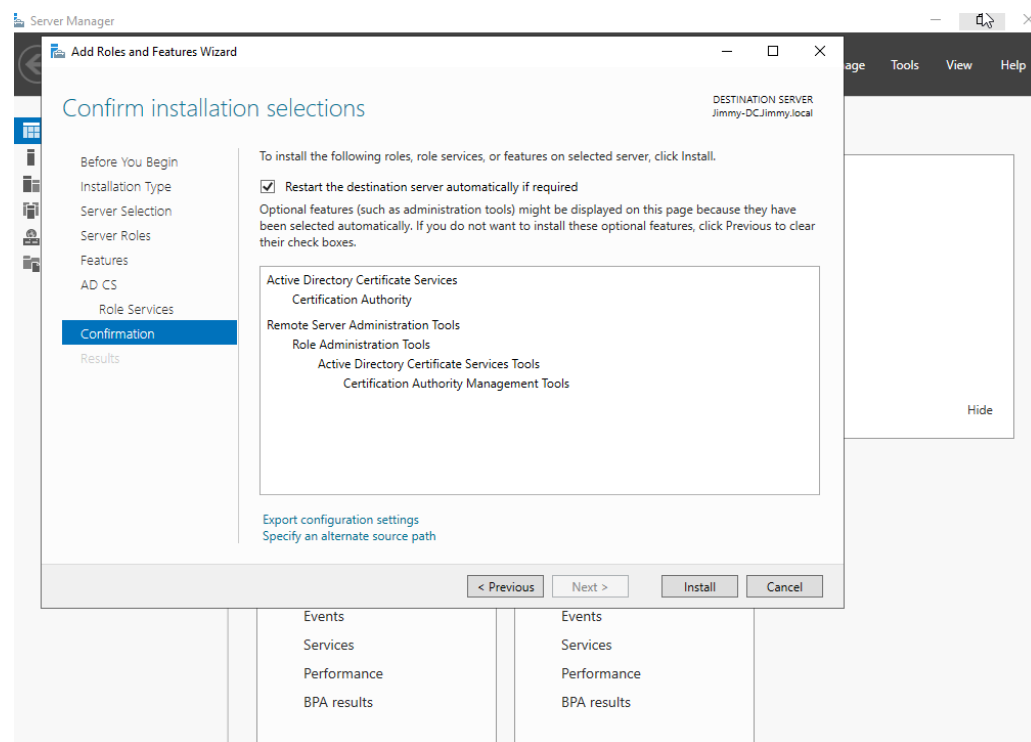
After the server role installation is complete, we would want to promote the Windows server to a Domain Controller. Since this is our first Domain Controller, we will create a new forest with a root domain name as “Jimmy.local” as our local Active Directory domain as shown below:

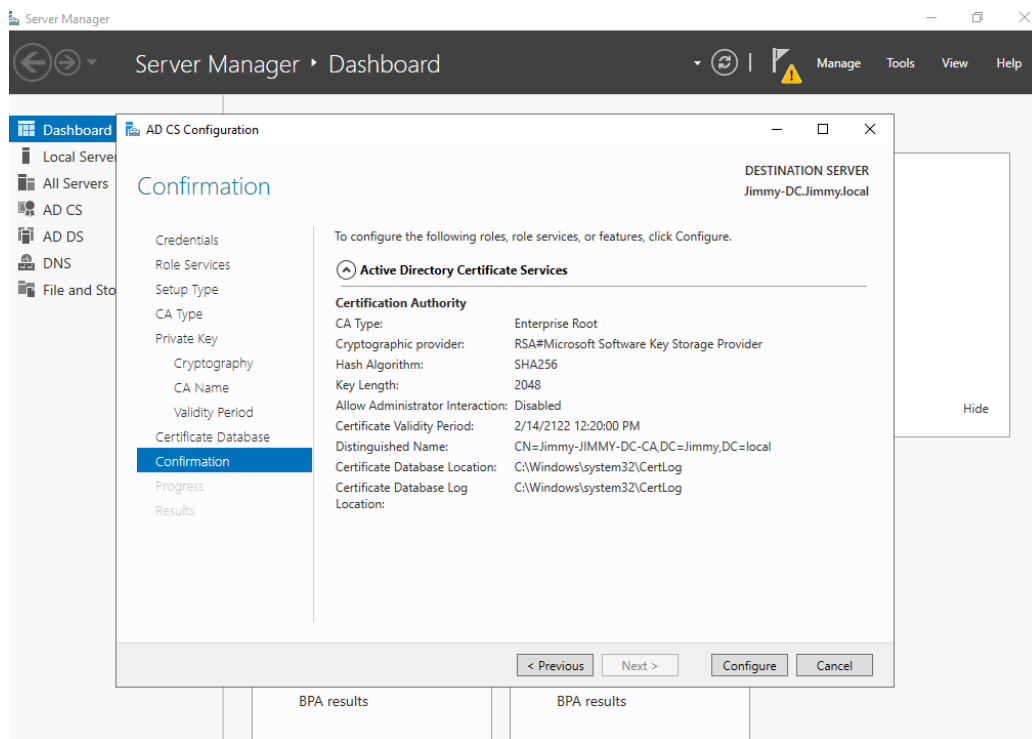
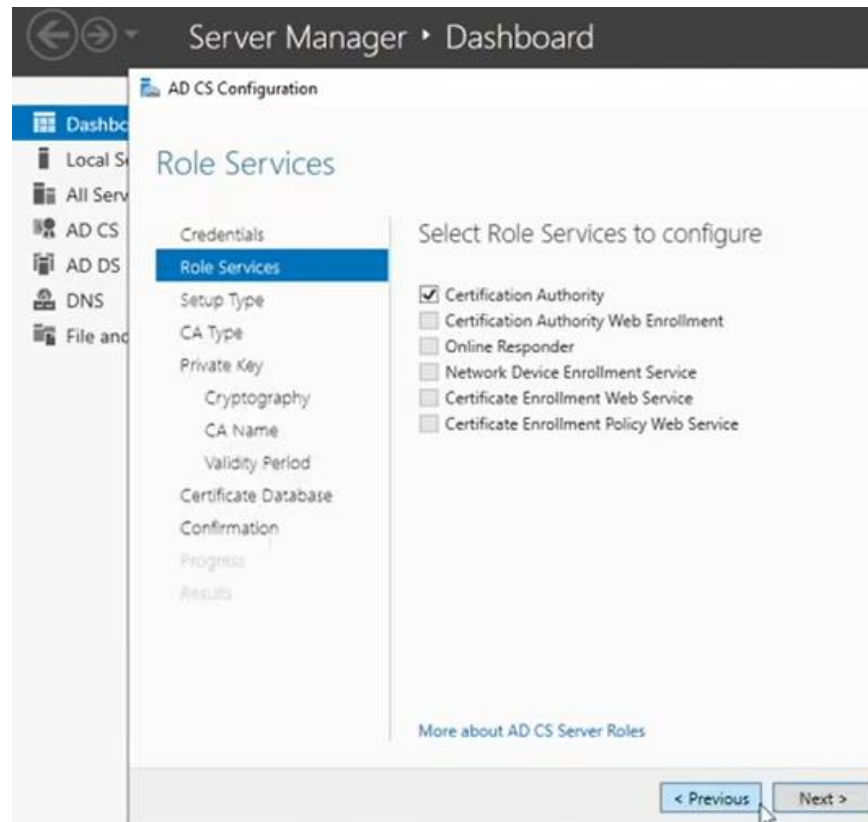


The Domain Controller will have DNS Server and Global catalog options checked by default as they are necessary for it to operate properly. The full configuration of the Domain Controller can be seen below:

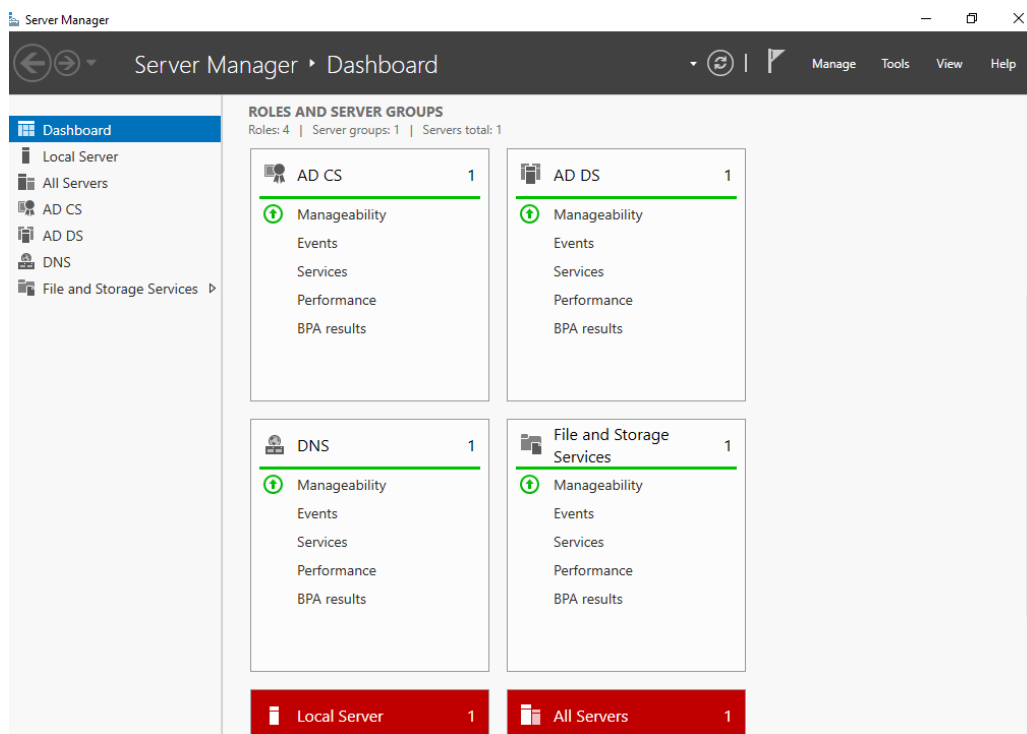


Next, we would want to go through the same process for setting up the Active Directory Certificate Services (AD CS). In the installation, we would want to check the box for Certificate Authority (CA) as that would allow the Domain Controller to act as a CA to validate certificates. The screenshots of settings that up can be seen below:

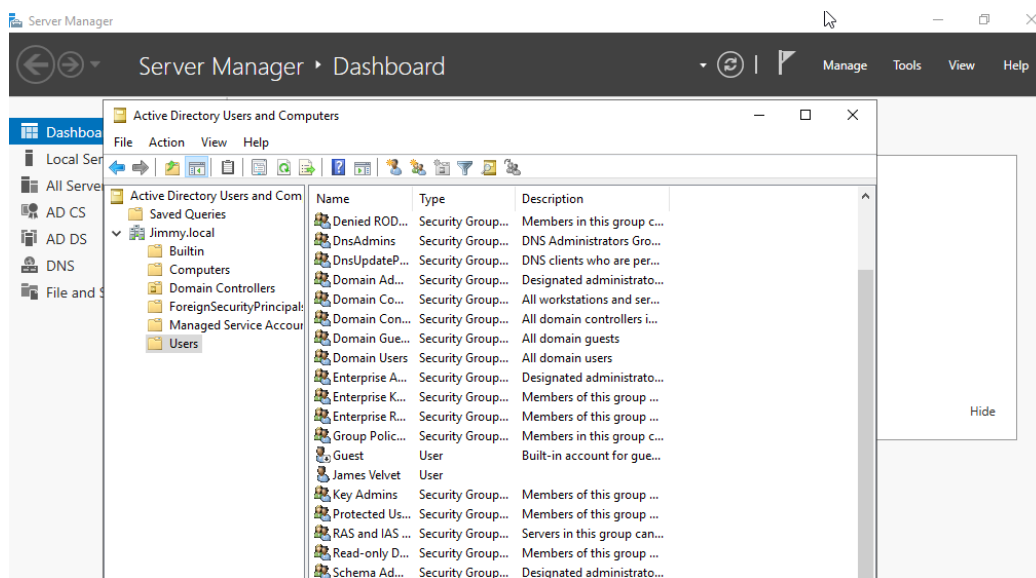




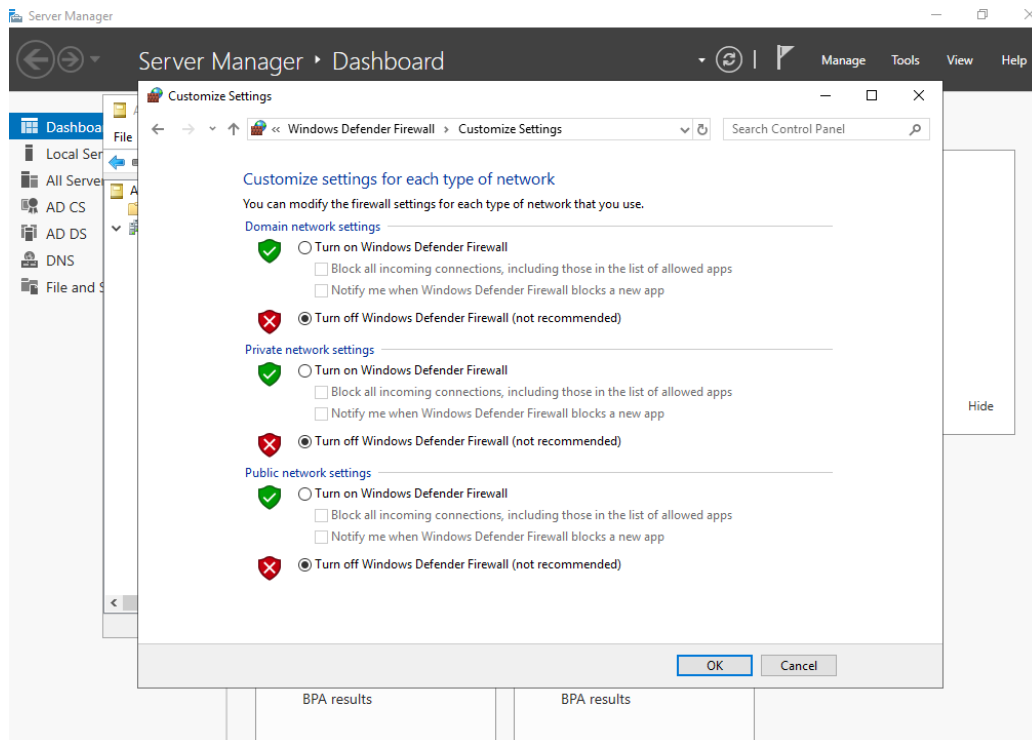
Once both AD DS and AD CS are properly installed and assigned to the now Windows Server Domain Controller, we should see this screenshot in our main dashboard below:



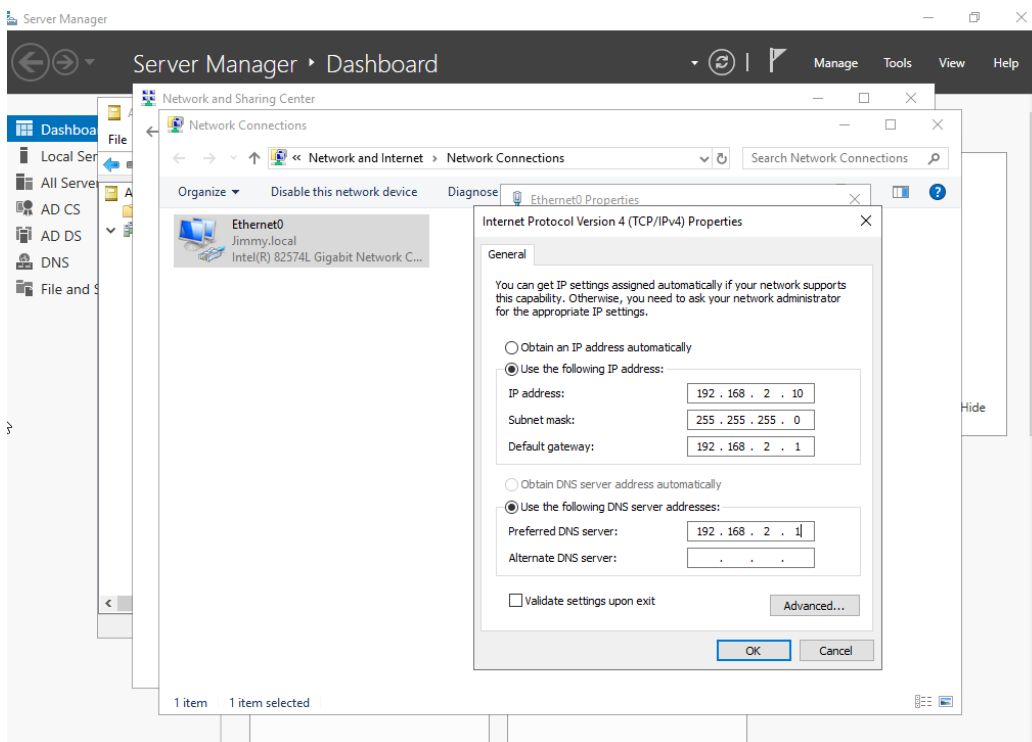
Next, we would want to create a new Windows user (named James Velvet) within the Users folder for our Active Directory as shown below:



Afterwards, we will disable all of the default Windows's firewall settings in order to maximize the amount of vulnerabilities we can generate from alerts and logs once we start detecting and monitoring them. The disabling of the Windows firewall can be seen below:

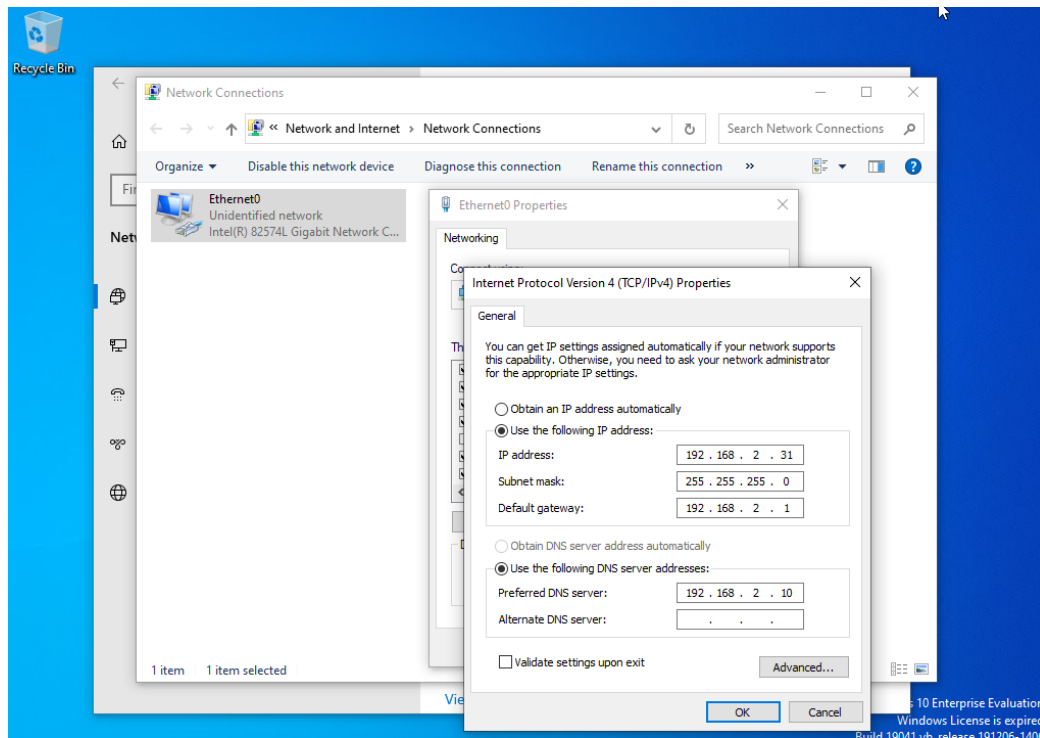


Lastly, we will assign the Domain Controller its network adapter configurations with 192.168.2.10 as its IP address, pfSense firewall as its default gateway and DNS server as seen below:

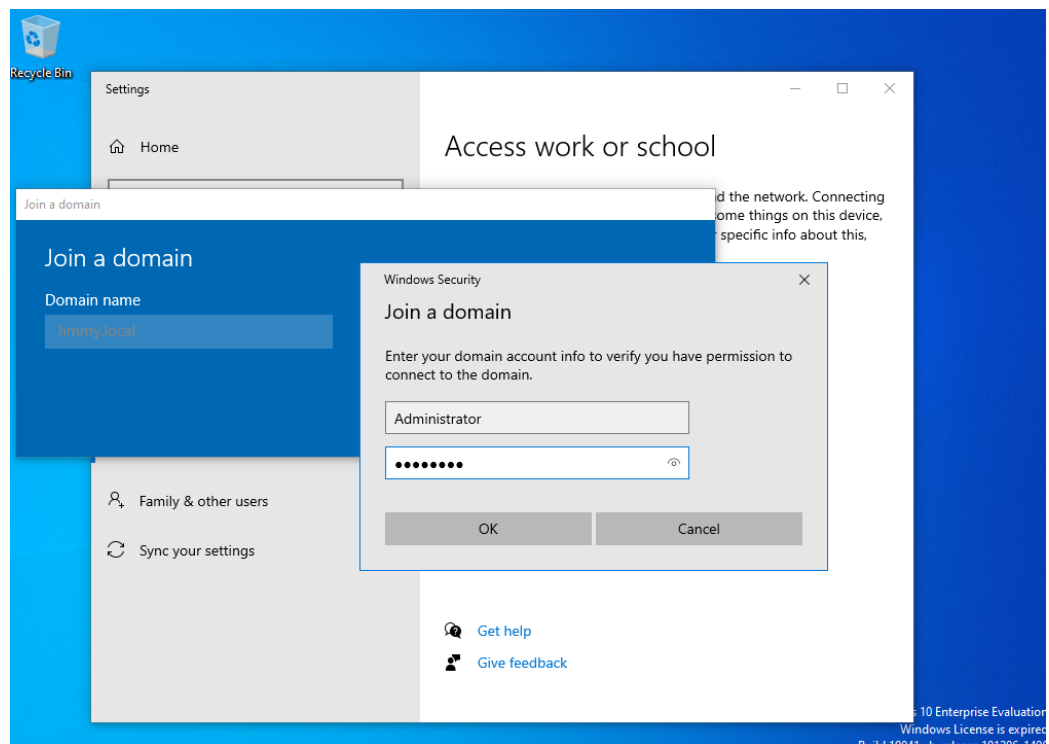
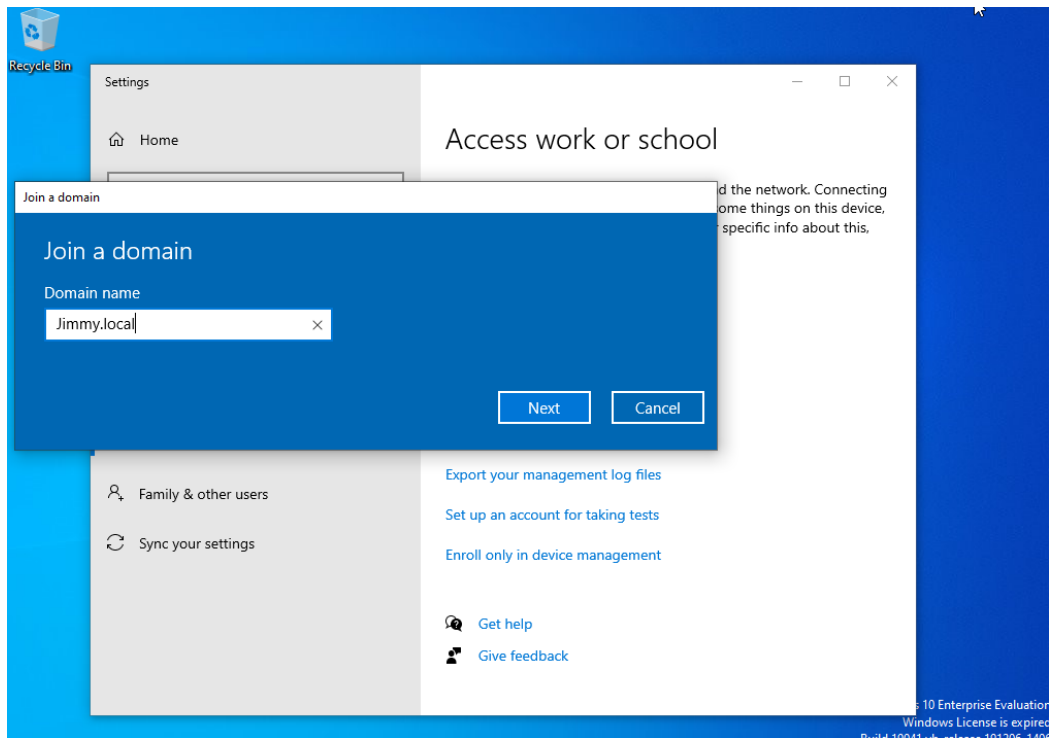


Adding the User to the Local Active Directory Domain

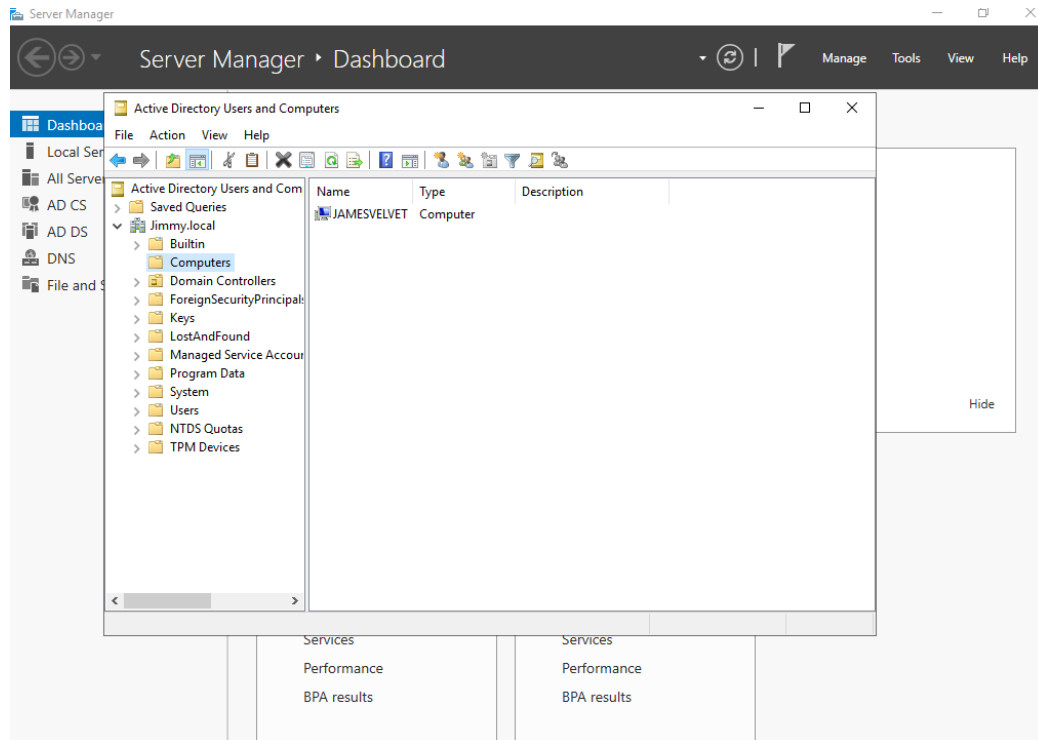
Once the virtual machine for the Windows user (James Velvet) is set and we have changed the PC name for that respective Windows user, we would want to first configure its network adapter settings with 192.168.2.31 as its IP address, pfSense as its default gateway, and the Domain Controller as its DNS server as seen below:



Afterwards, we will let the user join the local Active Directory domain (Jimmy.local) as seen below. The domain account used to verify is the Administrator account set up within the Windows Server at the very beginning of that section:



We can confirm that the Windows user (James Velvet) was able to join the local Active Directory domain by checking the Windows Server's Active Directory Computers folder as seen below:



Splunk Configuration

In this section, we will be setting up Splunk within an Ubuntu server. Once we setup an account for the server and accept the default settings for the Ubuntu server initial setup, we will reboot the Ubuntu server and be shown this screen after logging in with our credentials as shown below:

```

[ OK ] Finished Execute cloud user/final scripts.
[ OK ] Reached target Cloud-init target.

Password:

Login incorrect
splunk login: jimmy
Password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-60-generic x86_64)

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

System information as of Sun Feb 19 11:02:39 PM UTC 2023

System load:  0.55615234375   Processes:           239
Usage of /:   17.1% of 38.09GB Users logged in:      0
Memory usage: 9%             IPv4 address for ens32: 192.168.232.130
Swap usage:   0%

67 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

jimmy@splunk:~$

```

Next, we will install tasksel through the command from below, and through tasksel, we will install Ubuntu desktop (sudo tasksel install ubuntu-desktop) to obtain the GUI for the Ubuntu server:

```

[ OK ] Reached target Cloud-init target.

Password:

Login incorrect
splunk login: jimmy
Password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-60-generic x86_64)

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

System information as of Sun Feb 19 11:02:39 PM UTC 2023

System load:  0.55615234375   Processes:           239
Usage of /:   17.1% of 38.09GB Users logged in:      0
Memory usage: 9%             IPv4 address for ens32: 192.168.232.130
Swap usage:   0%

67 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

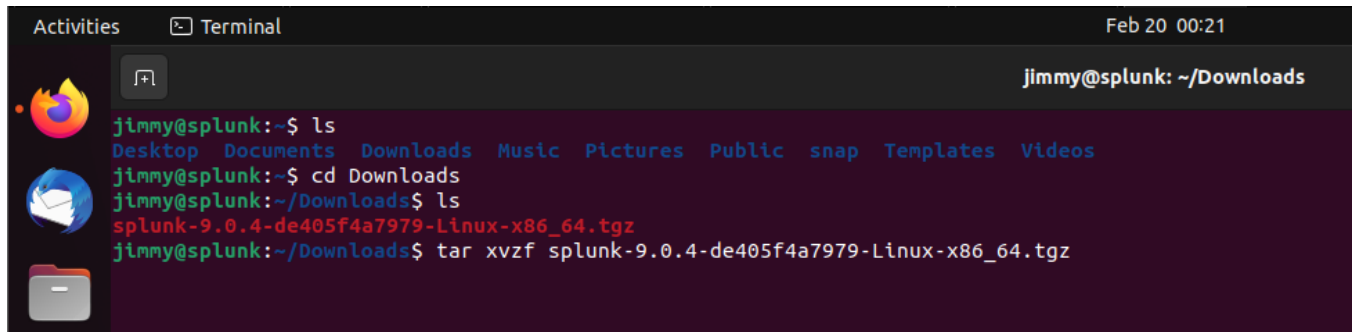
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

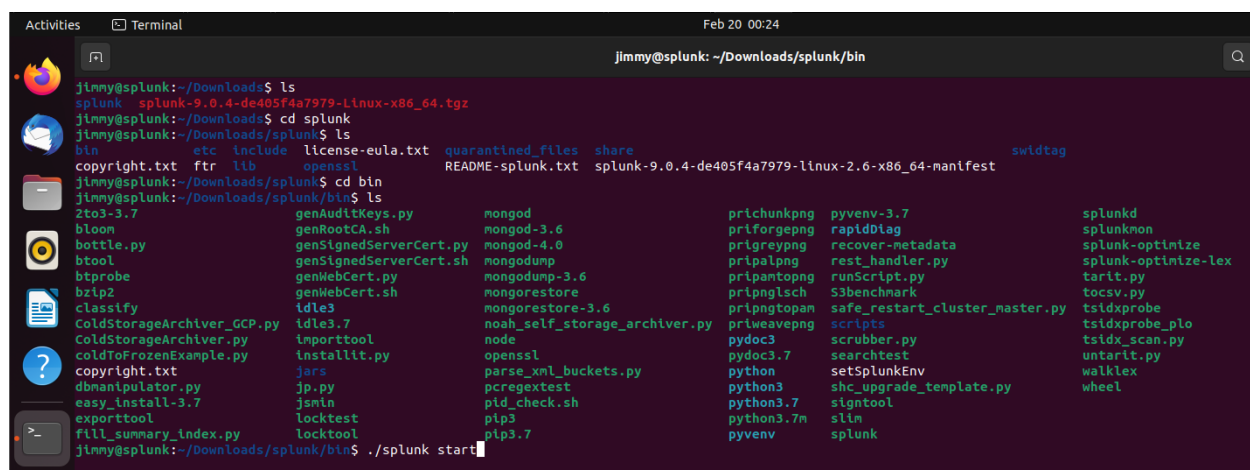
jimmy@splunk:~$ sudo apt install tasksel
[sudo] password for jimmy:

```

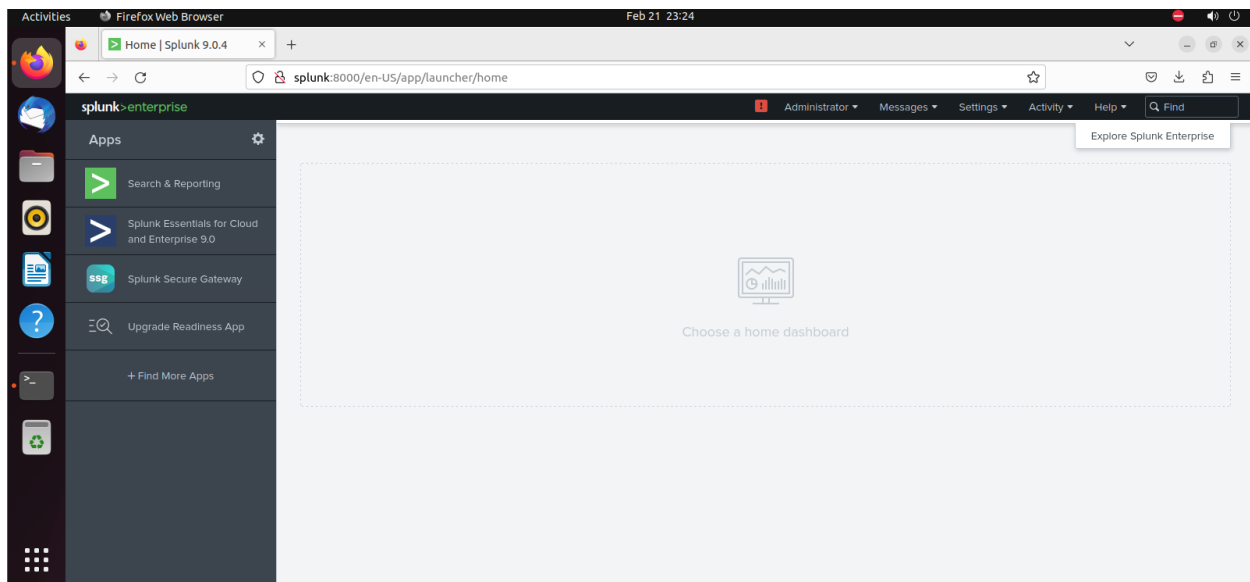
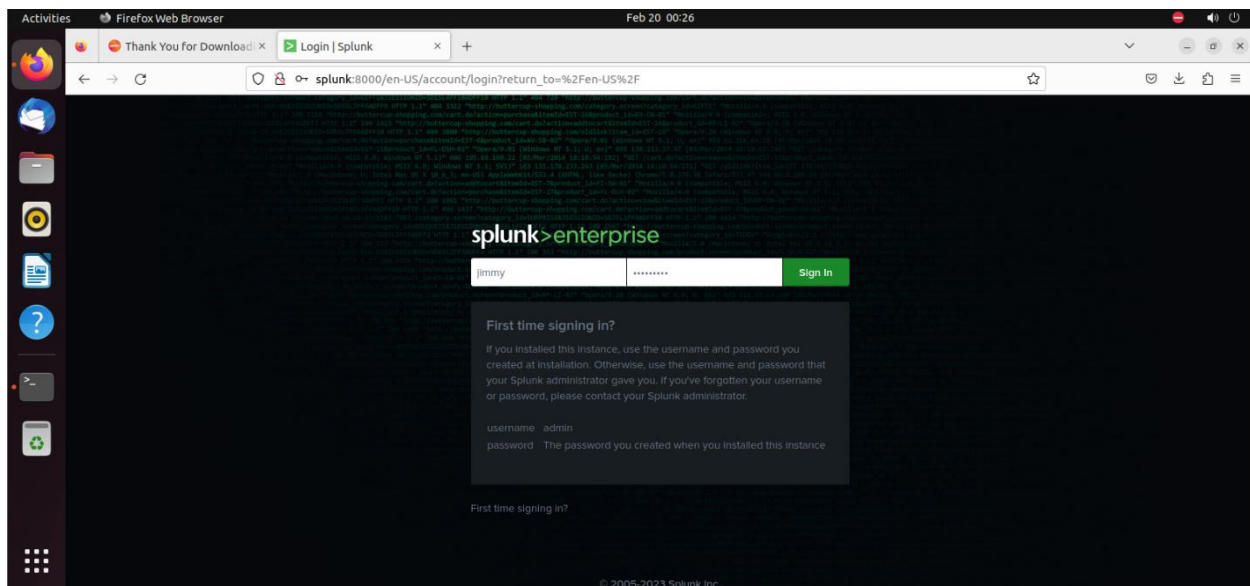
After rebooting the server once the tasksel installation is complete, we should be able to obtain the GUI for the Ubuntu server. Once we login to the server, we will use a web browser to download a free trial of Splunk Enterprise on their official website. From there, we can open a terminal and head over to the Downloads directory to view the downloaded Splunk file as seen below:

A terminal window titled 'Terminal' with the prompt 'jimmy@splunk: ~/Downloads'. The user enters 'ls' and the output shows a list of directories: Desktop, Documents, Downloads, Music, Pictures, Public, snap, Templates, and Videos. Then, the user enters 'cd Downloads' and 'ls' again, showing a file named 'splunk-9.0.4-de405f4a7979-Linux-x86_64.tgz'. Finally, the user enters 'tar xvzf splunk-9.0.4-de405f4a7979-Linux-x86_64.tgz'.

As seen from above, we will utilize the tar command to obtain the actual Splunk application within another directory as seen below:

A terminal window titled 'Terminal' with the prompt 'jimmy@splunk: ~/Downloads/splunk/bin'. The user enters 'ls' and the output shows a long list of files and directories, including bin, etc, include, license-eula.txt, quarantined_files, share, swidtag, README-splunk.txt, and splunk-9.0.4-de405f4a7979-linux-2.6-x86_64-manifest. Then, the user enters 'cd bin' and 'ls' again, showing a long list of files and directories, including 2to3-3.7, bloom, bottle.py, btool, btprobe, bz2, classify, ColdStorageArchiver_GCP.py, ColdStorageArchiver.py, ColdToFrozenExample.py, copyright.txt, dbmanipulator.py, easy_install-3.7, exporttool, fill_summary_index.py, genAuditKeys.py, genRootCA.sh, genSignedServerCert.py, genSignedServerCert.sh, genWebCert.py, genWebCert.sh, idle3, idle3.7, importtool, installit.py, jars, jp.py, jsmn, locktest, locktool, mongod, mongod-3.6, mongod-4.0, mongodump, mongodump-3.6, mongorestore, mongorestore-3.6, noah_self_storage_archiver.py, node, openssl, parse_xml_buckets.py, pcregextest, pid_check.sh, pip3, pip3.7, prichunkpng, priforgepng, prigraypng, pripalpng, pripamtopng, pripnglsch, pripngtopam, priweavepng, pydoc3, pydoc3.7, python, python3, python3.7, python3.7m, pyvenv, splunkd, splunkmon, splunk-optimize, splunk-optimize-lex, tarit.py, tocsv.py, tsidxprobe, tsidxprobe_plo, tsidx_scan.py, untarit.py, walklex, and wheel.

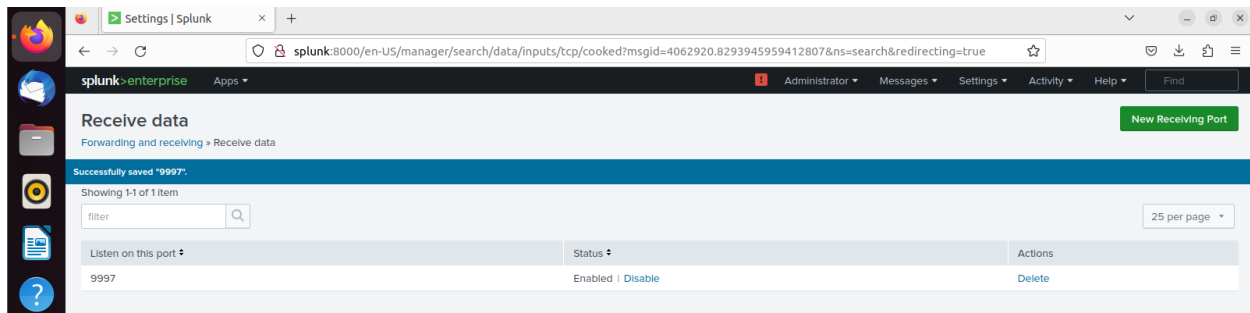
We will then start a Splunk instance with the `./splunk start` command and make a Splunk administrator account. Once the account has been made, we can then access Splunk on the web browser through port 8000, which the URL would be <http://splunk:8000>. The following pages will load up when entering that URL and logging in with the administrator account, which the user now has access to Splunk:



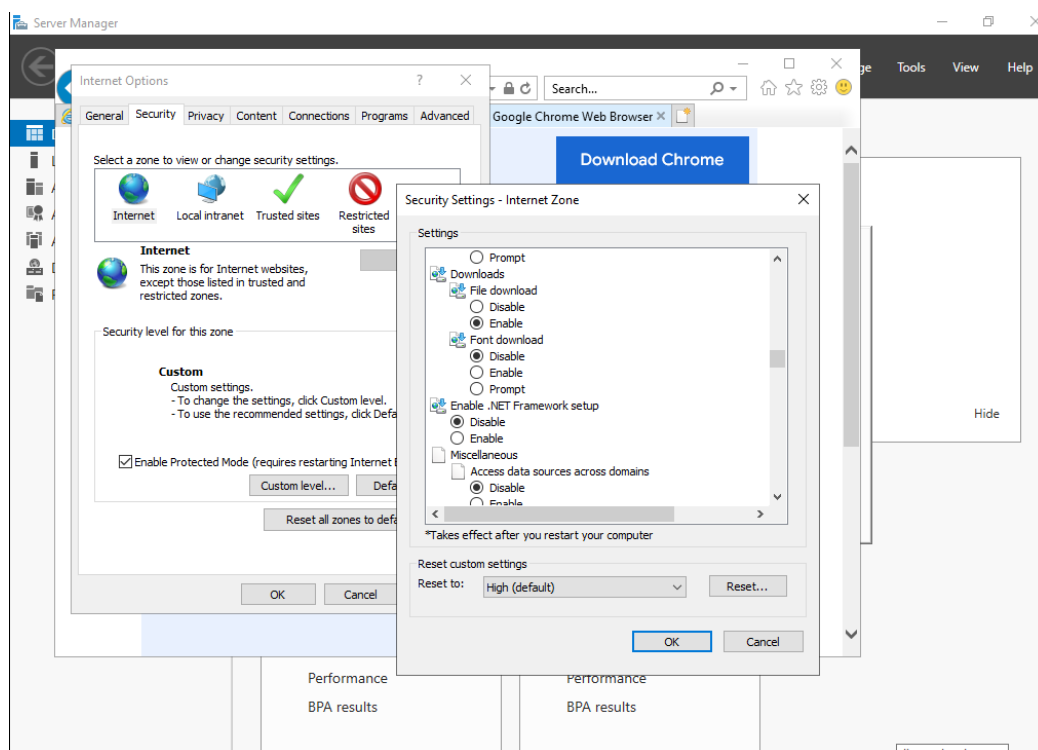
Installing Universal Forwarder on Windows Server

Splunk utilizes a mechanism called the universal forwarder in order to log activities from different endpoints on a network. The universal forwarder will be connected to the victim network for Splunk to monitor activities from there, which the universal forwarder will be installed on the Windows Server.

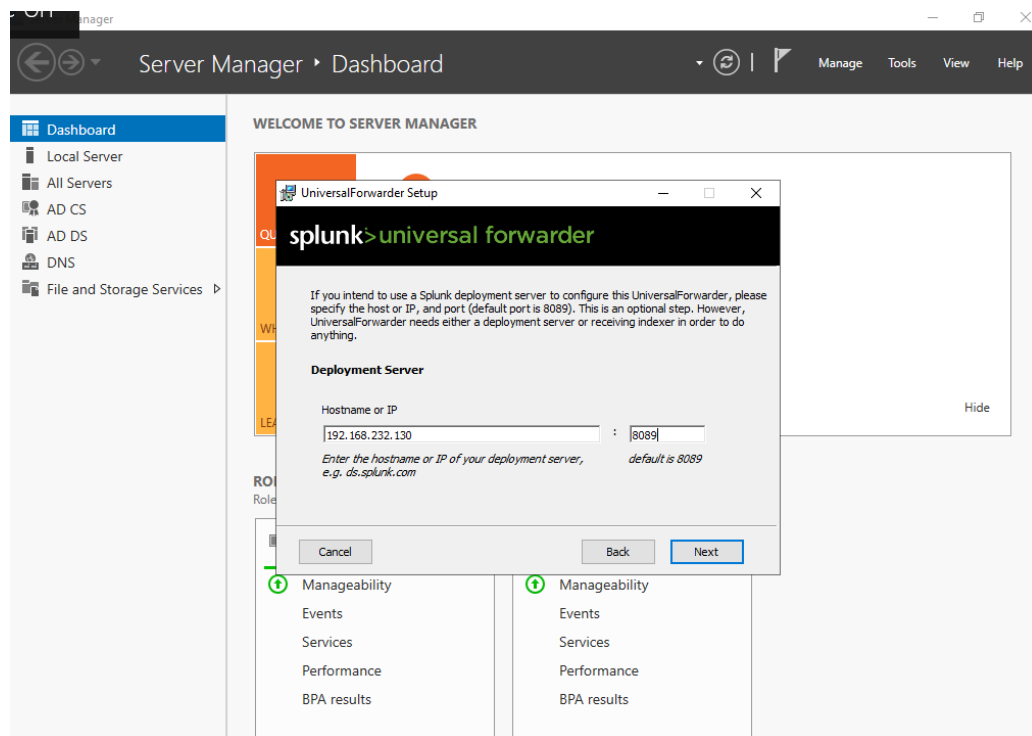
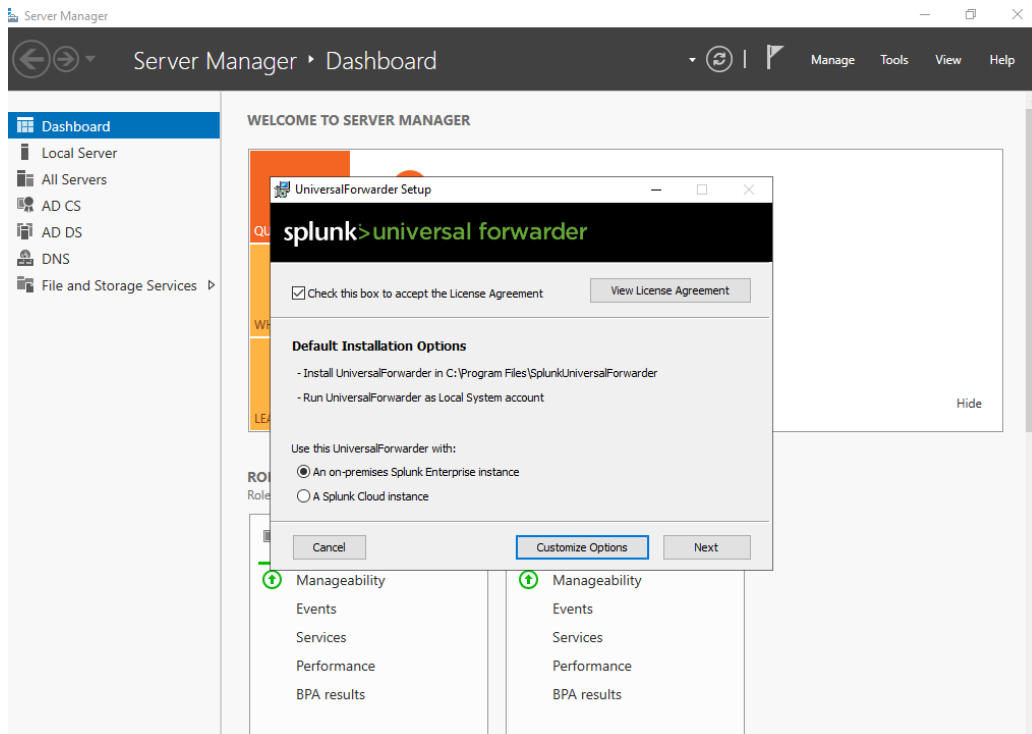
Before we install the universal forwarder, we will first setup a receiving port for the Splunk instance to receive data from the universal forwarder. We will setup Splunk to receive data from port 9997 as that is the default port for Splunk to listen for universal forwarders as shown below:

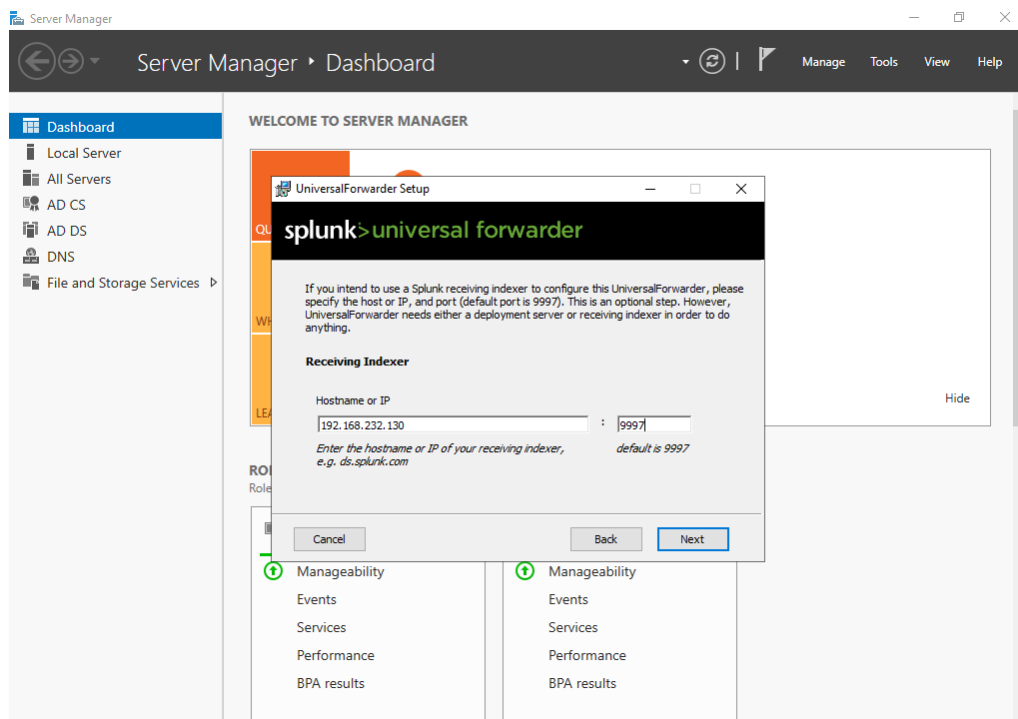


Afterwards, we can start downloading the universal forwarder on the Windows Server. I had to enable file downloads on Internet Explorer as it is by default disabled before downloading the universal forwarder as shown below:

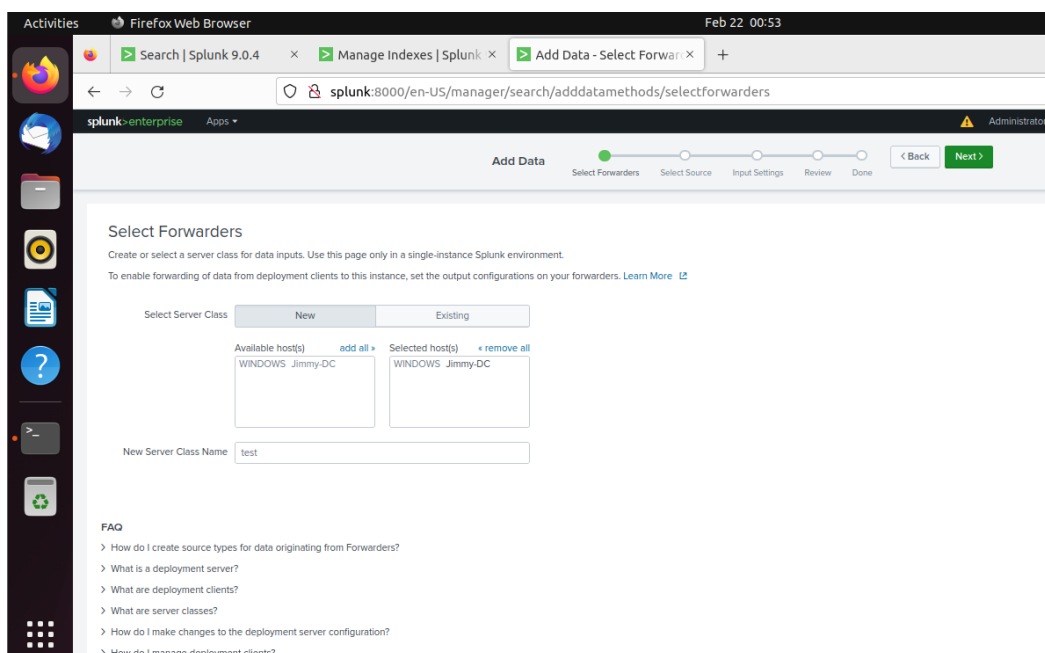


Once the download is complete, we can start setting up the configurations for the Splunk universal forwarder. We will configure the universal forwarder to use an on premise Splunk Enterprise instance with the deployment server and receiving indexer being the IP address of the Splunk instance found on the Ubuntu server as shown below:



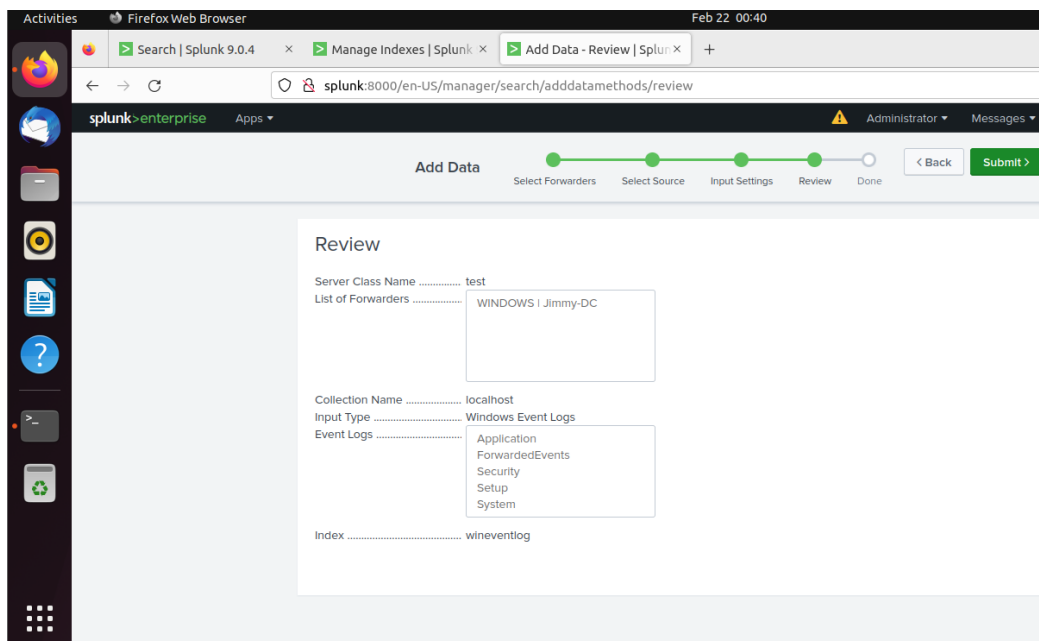


After the universal forwarder's installation is complete, we will now complete the setup for Splunk to receive data from the universal forwarder. If the configuration is correct, we should see our Domain Controller on the list of forwarders as that is where our universal forwarder is installed as seen below:



From there, we will create an index called wineventlog where all of the data Splunk received from the universal forwarder will be stored in. As the name implies, the index will store all data related to local Windows events such as security logs, system logs,

etc. The final configuration for Splunk to monitor and store data logs from the universal forwarder can be seen below:



With Splunk now configured to listen to port 9997 from the universal forwarder installed on the Windows Server, we are now able to use Splunk for monitoring and detecting log events occurring from the victim network. As a test, we decided to sign out of the Windows Server administrator account and sign back in to see if Splunk was able to monitor that event.

The two images below show the wineventlog index set up within Splunk, where the first image was taken prior to the test of signing out of the Windows Server and the second image was take after the test:

The screenshot shows the 'Manage Indexes' page in the Splunk web interface. The table below lists the configured indexes, including the 'wineventlog' index.

Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen Path	Status
__audit	Edit Delete Disable	Events	system	2 MB	488.28 GB	10.4K	2 days ago	a few seconds ago	\$SPLUNK_DB/_audit/db	N/A	✓ Enabled
__configtracker	Edit Delete Disable	Events	system	2 MB	488.28 GB	186	2 days ago	2 days ago	\$SPLUNK_DB/_configtracker/db	N/A	✓ Enabled
__internal	Edit Delete Disable	Events	system	2 MB	488.28 GB	19.8K	2 days ago	a few seconds ago	\$SPLUNK_DB/_internaldb/db	N/A	✓ Enabled
__introspection	Edit Delete Disable	Events	system	5 MB	488.28 GB	2.62K	2 days ago	a few seconds ago	\$SPLUNK_DB/_introspection/db	N/A	✓ Enabled
__metrics	Edit Delete Disable	Metrics	system	4 MB	488.28 GB	14.5K	2 days ago	a few seconds ago	\$SPLUNK_DB/_metrics/db	N/A	✓ Enabled
__metrics_rollup	Edit Delete Disable	Metrics	system	1 MB	488.28 GB	0			\$SPLUNK_DB/_metrics_rollup/db	N/A	✓ Enabled
__telemetry	Edit Delete Disable	Events	system	1 MB	488.28 GB	2	2 days ago	3 minutes ago	\$SPLUNK_DB/_telemetry/db	N/A	✓ Enabled
__thebucket	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/_fishbucket/db	N/A	✓ Enabled
history	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/historydb/db	N/A	✓ Enabled
main	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/defaultdb/db	N/A	✓ Enabled
splunklogger	Edit Delete Enable	Events	system	0 B	488.28 GB	0			\$SPLUNK_DB/splunklogger/db	N/A	✗ Disabled
summary	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/summarydb/db	N/A	✓ Enabled
wineventlog	Edit Delete Disable	Events	search	1 MB	500 GB	0			\$SPLUNK_DB/wineventlog/db	N/A	✓ Enabled

Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen Path	Status
__audit	Edit Delete Disable	Events	system	3 MB	488.28 GB	26.8K	2 days ago	a few seconds ago	\$SPLUNK_DB/audit/db	N/A	✓ Enabled
__configtracker	Edit Delete Disable	Events	system	2 MB	488.28 GB	231	2 days ago	14 minutes ago	\$SPLUNK_DB/_configtracker/db	N/A	✓ Enabled
__internal	Edit Delete Disable	Events	system	6 MB	488.28 GB	85.9K	2 days ago	in 3 hours	\$SPLUNK_DB/_internaldb/db	N/A	✓ Enabled
__introspection	Edit Delete Disable	Events	system	13 MB	488.28 GB	9.2K	2 days ago	a few seconds ago	\$SPLUNK_DB/_introspection/db	N/A	✓ Enabled
__metrics	Edit Delete Disable	Metrics	system	6 MB	488.28 GB	50.4K	2 days ago	a few seconds ago	\$SPLUNK_DB/_metrics/db	N/A	✓ Enabled
__metrics_rollup	Edit Delete Disable	Metrics	system	1 MB	488.28 GB	0			\$SPLUNK_DB/_metrics_rollup/db	N/A	✓ Enabled
__telemetry	Edit Delete Disable	Events	system	1 MB	488.28 GB	4	2 days ago	an hour ago	\$SPLUNK_DB/_telemetry/db	N/A	✓ Enabled
__the_fishbucket	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/_the_fishbucket/db	N/A	✓ Enabled
history	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/historydb/db	N/A	✓ Enabled
main	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/defaultdb/db	N/A	✓ Enabled
splunklogger	Edit Delete Enable	Events	system	0 B	488.28 GB	0			\$SPLUNK_DB/splunklogger/db	N/A	✗ Disabled
summary	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/summarydb/db	N/A	✓ Enabled
wineventlog	Edit Delete Disable	Events	search	2 MB	500 GB	718K	8 days ago	in 3 hours	\$SPLUNK_DB/wineventlog/db	N/A	✓ Enabled

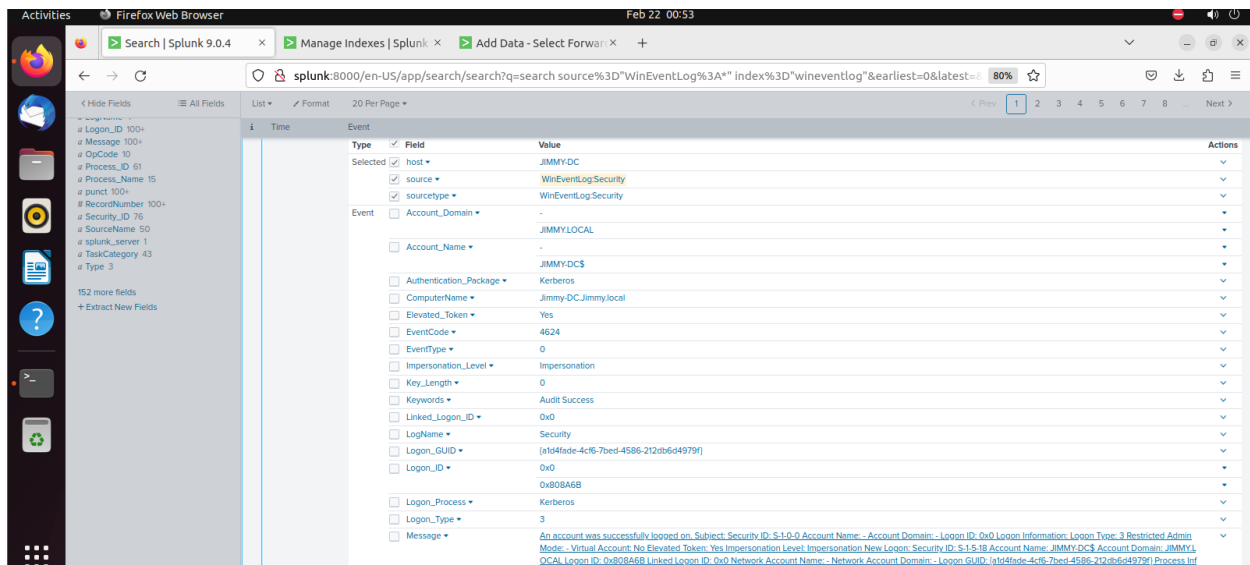
As we can see in the second image above, the test was successful in which the wineventlog index was updated with the latest event now has a entry compared to being blank before and the current size of the index went up from 1 MB to 2 MB.

We can conduct a search on the wineventlog index and view some of the logs recorded by Splunk. The next two images show two separate security logs recording the events of us signing in and out of the Windows Server:

Activities

Firefox Web Browser

Feb 28, 00:50



Next Steps

As this lab documentation was primarily focused on setting up and configuring different aspects of the virtual network as a whole, utilizing the security tools and programs within this network was limited and a secondary focus. However, with everything in our virtual home lab network properly set up, we can now conduct all sorts of attacks for our SIEM tools to detect, or even produce innocent, but purposeful events for us to practice analyzing data through auditing and reading logs from Splunk or Security Onion. Through these series of sandboxing, configuring defenses, and producing controlled attacks on our virtual network, one can learn all sorts of new skills from configuring rule detection to analyzing attack vectors of a network.

Some future plans to continue exploring for this home lab include:

- Explore and understand how Splunk works; configure, tune, and test different rules and detection methods Splunk has to offer
- Design more firewall rules and explore the capabilities of what pfSense has to offer for network security
- Learn more about Active Directory and how security can be applied in such environments; test and configure more on the Domain Controller