

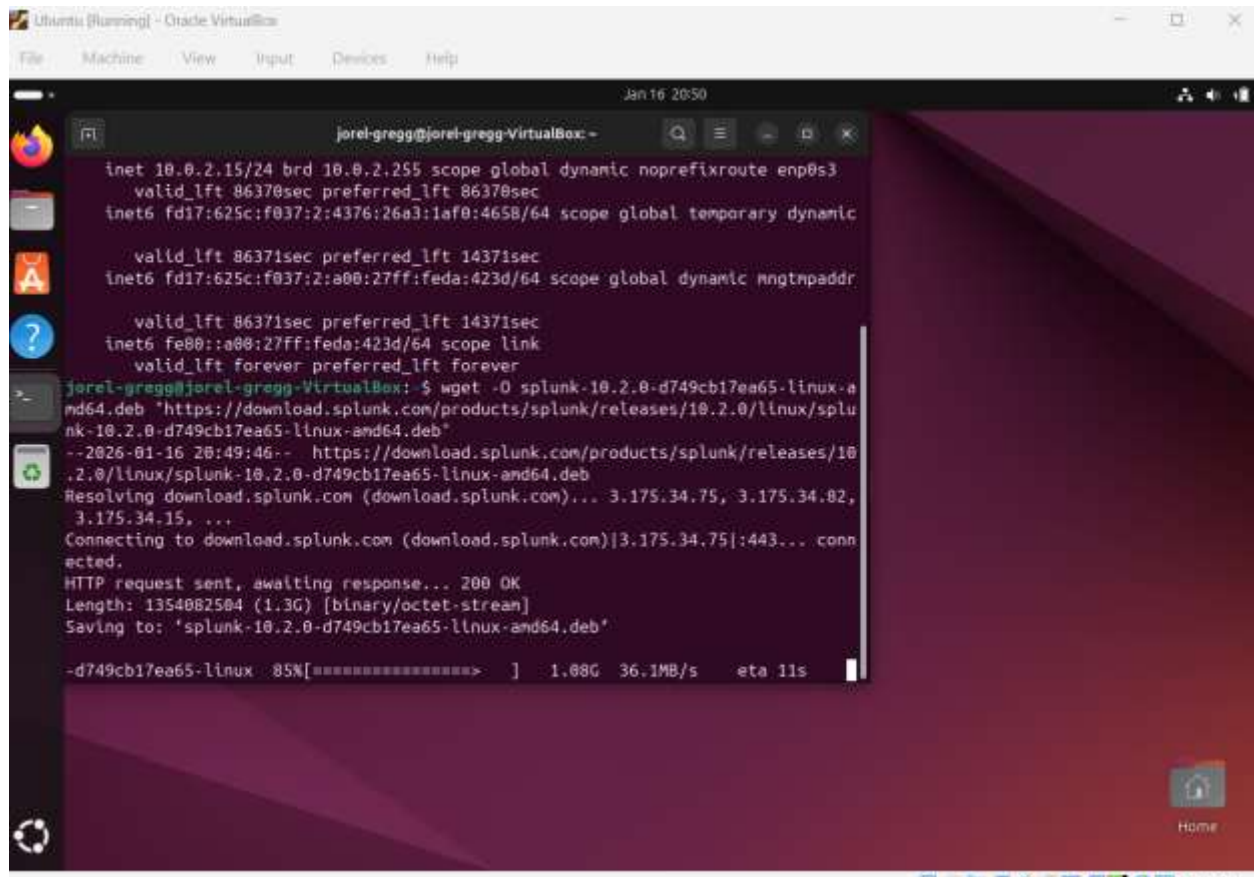
Jorel Gregg

Basic SPLUNK Home Lab

1/16/26

This will be my first time installing and running SPLUNK on my home virtual lab through VirtualBox. I will use the same Kali-Linux and Ubuntu machines I used in my Wireshark lab.

1. First, I created an account in Splunk and then copied the wget link for the .deb installation.



```
jorel-gregg@jorel-gregg-VirtualBox:~$ cat /etc/network/interfaces
auto eth0
iface eth0 inet static
    address 10.0.2.15
    netmask 255.255.255.0
    gateway 10.0.2.1
    dns-nameservers 8.8.8.8

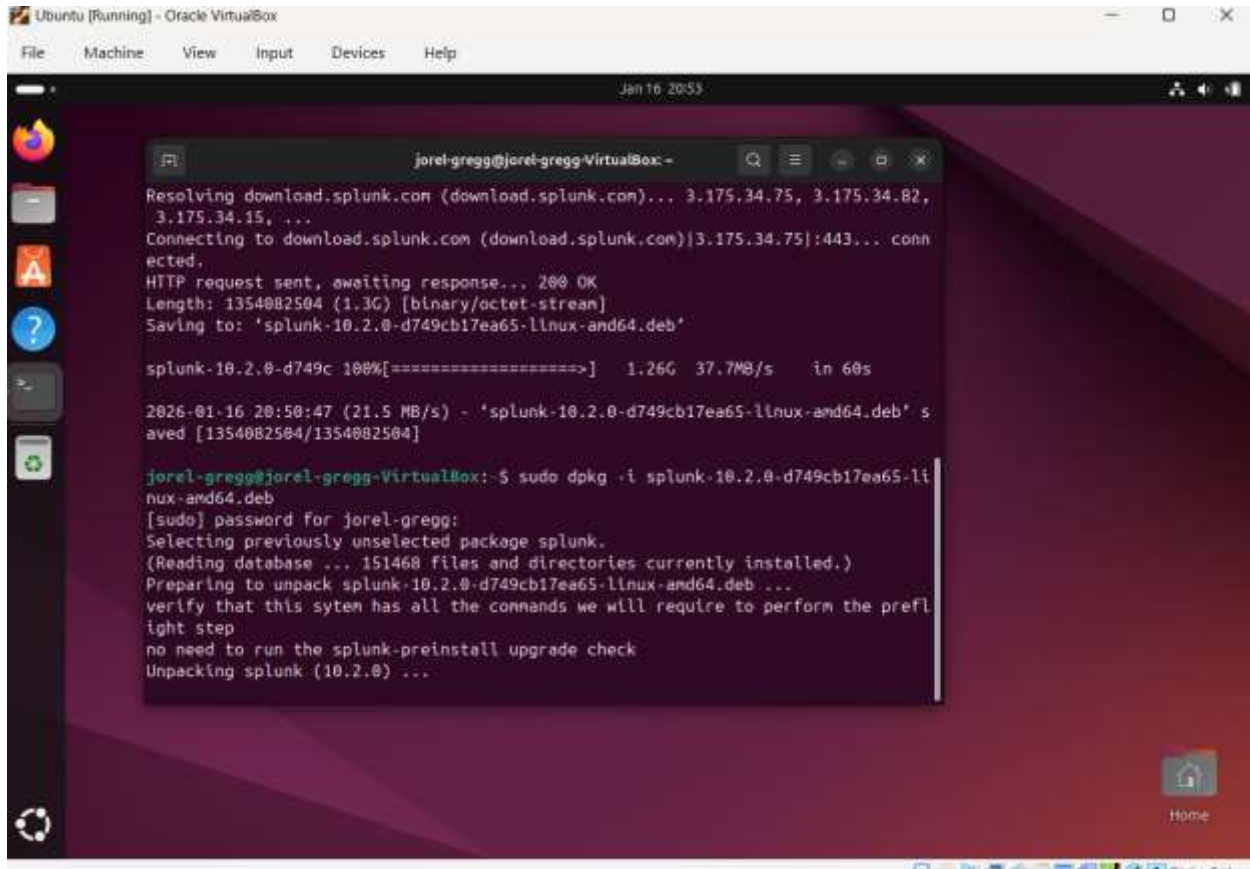
jorel-gregg@jorel-gregg-VirtualBox:~$ cat /etc/network/interfaces
auto eth0
iface eth0 inet static
    address 10.0.2.15
    netmask 255.255.255.0
    gateway 10.0.2.1
    dns-nameservers 8.8.8.8

jorel-gregg@jorel-gregg-VirtualBox:~$ cat /etc/network/interfaces
auto eth0
iface eth0 inet static
    address 10.0.2.15
    netmask 255.255.255.0
    gateway 10.0.2.1
    dns-nameservers 8.8.8.8

jorel-gregg@jorel-gregg-VirtualBox:~$ wget -O splunk-10.2.0-d749cb17ea65-linux-amd64.deb "https://download.splunk.com/products/splunk/releases/10.2.0/linux/splunk-10.2.0-d749cb17ea65-linux-amd64.deb"
--2026-01-16 20:49:46-- https://download.splunk.com/products/splunk/releases/10.2.0/linux/splunk-10.2.0-d749cb17ea65-linux-amd64.deb
Resolving download.splunk.com (download.splunk.com)... 3.175.34.75, 3.175.34.82, 3.175.34.15, ...
Connecting to download.splunk.com (download.splunk.com)[3.175.34.75]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1354082504 (1.3G) [binary/octet-stream]
Saving to: 'splunk-10.2.0-d749cb17ea65-linux-amd64.deb'

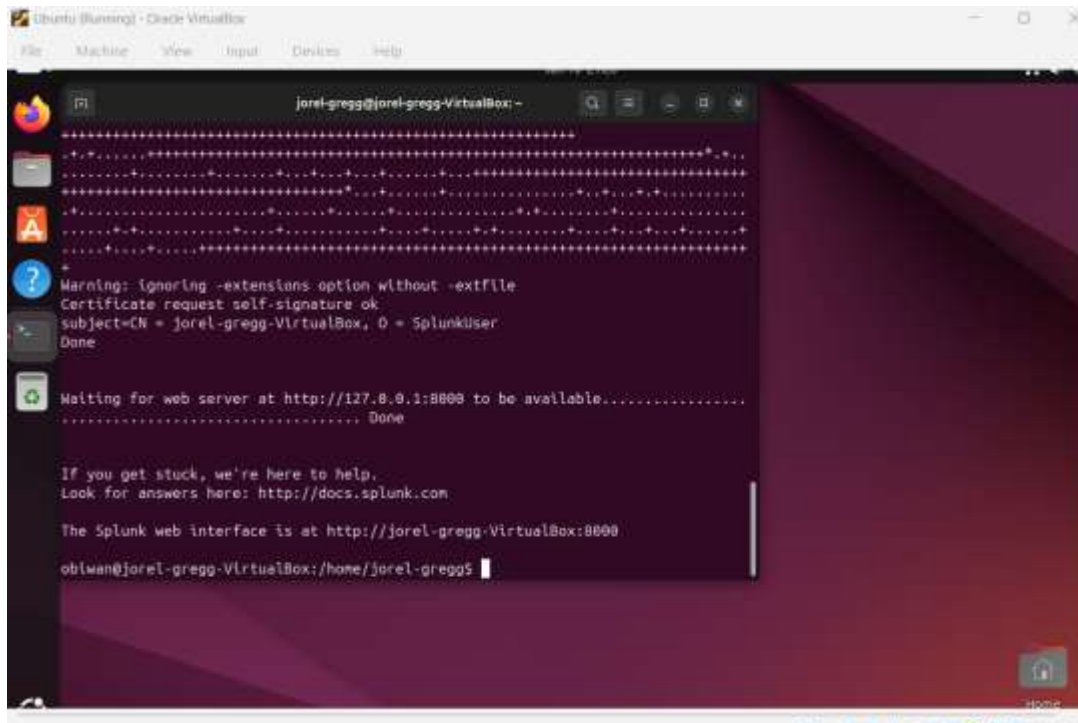
-d749cb17ea65-linux 85%[=====> ] 1.08G 36.1MB/s eta 11s
```

2. I then used the following command to install Splunk on the Ubuntu machine.
 - a. `sudo dpkg -i splunk-xxx.deb`

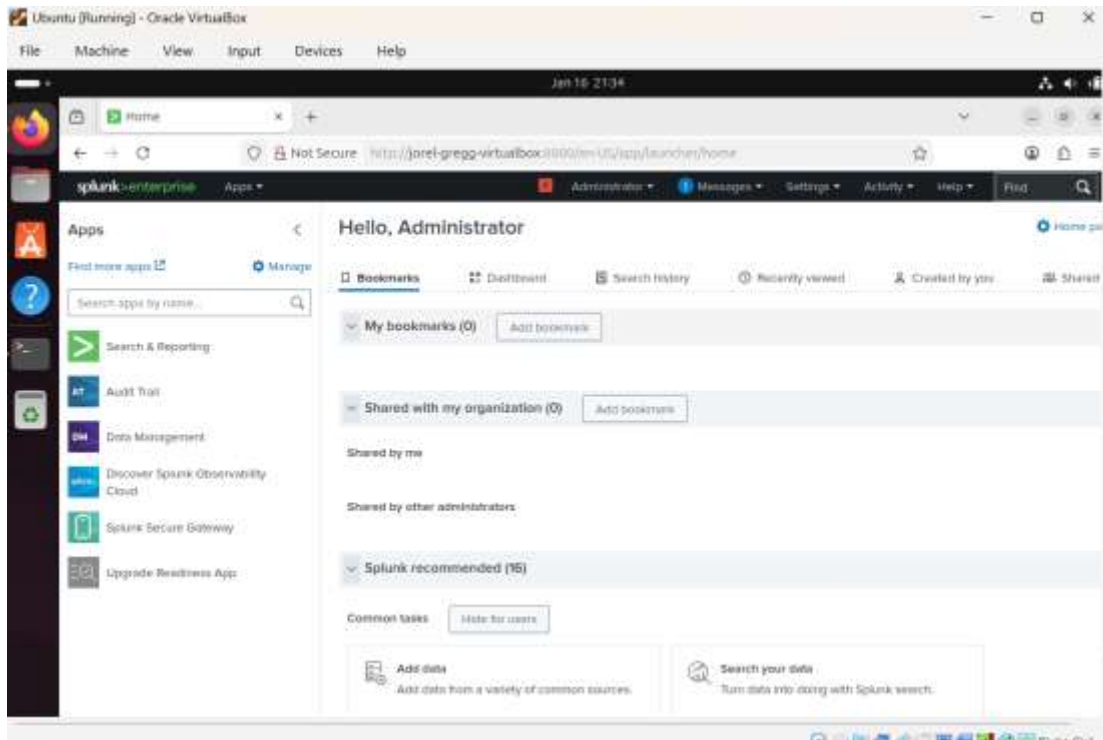


```
jorel-gregg@jorel-gregg-VirtualBox: ~$ curl -O https://download.splunk.com/dist/10.2.0/splunk-10.2.0-d749cb17ea65-linux-amd64.deb
jorel-gregg@jorel-gregg-VirtualBox: ~$ ls -l splunk-10.2.0-d749cb17ea65-linux-amd64.deb
-rw-r--r-- 1 jorel-gregg jorel-gregg 1354082504 Jan 16 20:50 splunk-10.2.0-d749cb17ea65-linux-amd64.deb
jorel-gregg@jorel-gregg-VirtualBox: ~$ sudo dpkg -i splunk-10.2.0-d749cb17ea65-linux-amd64.deb
[sudo] password for jorel-gregg:
Selecting previously unselected package splunk.
(Reading database ... 151468 files and directories currently installed.)
Preparing to unpack splunk-10.2.0-d749cb17ea65-linux-amd64.deb ...
verify that this system has all the commands we will require to perform the preflight step
no need to run the splunk-preinstall upgrade check
Unpacking splunk (10.2.0) ...
```

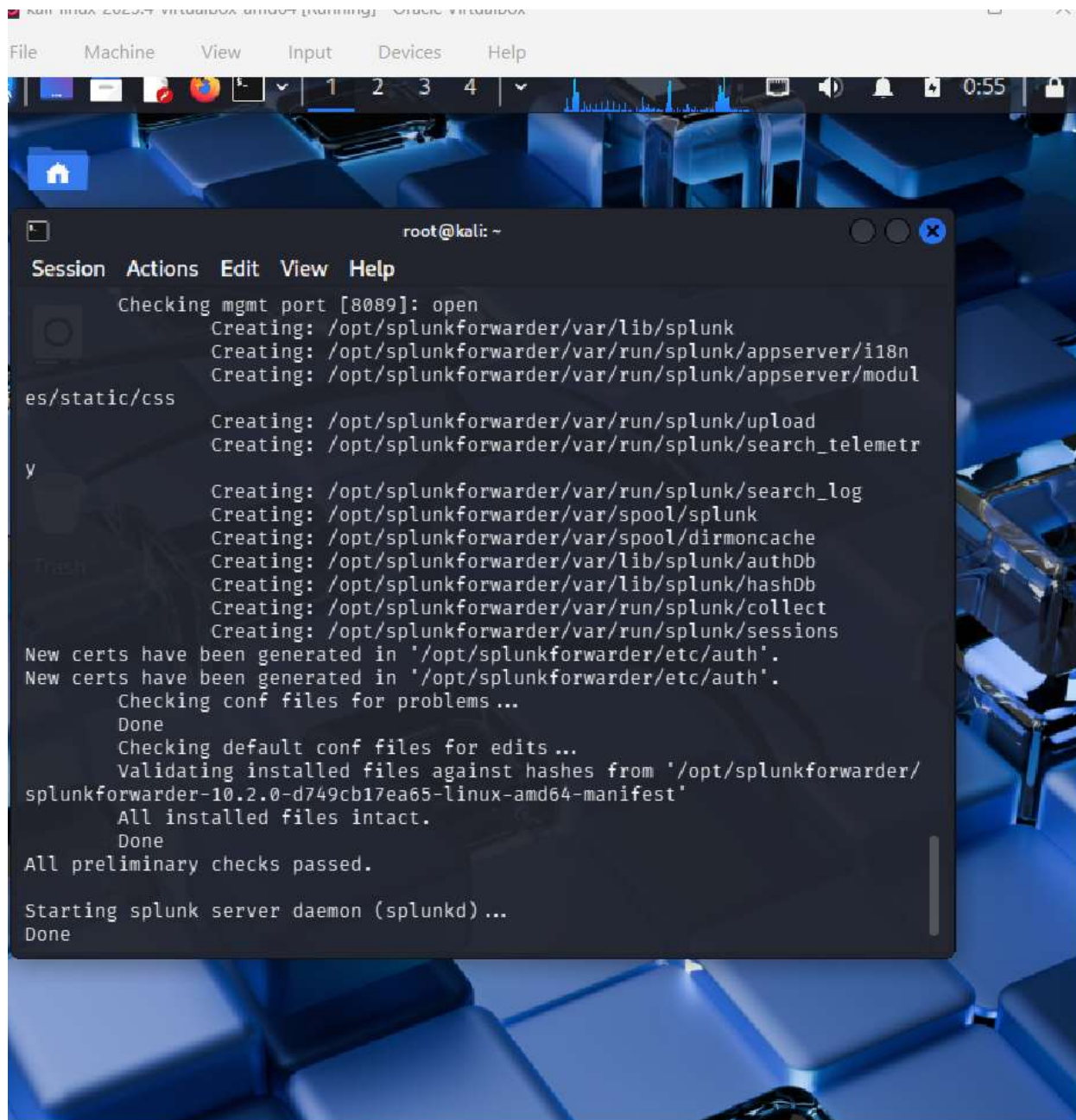
- The next step was to launch Splunk which, after a lot of failure and research, I found I had to do through a non-root user and set their permissions to run the program.



- Once I knew it was up and running, I needed to reach the Splunk interface in my web browser.



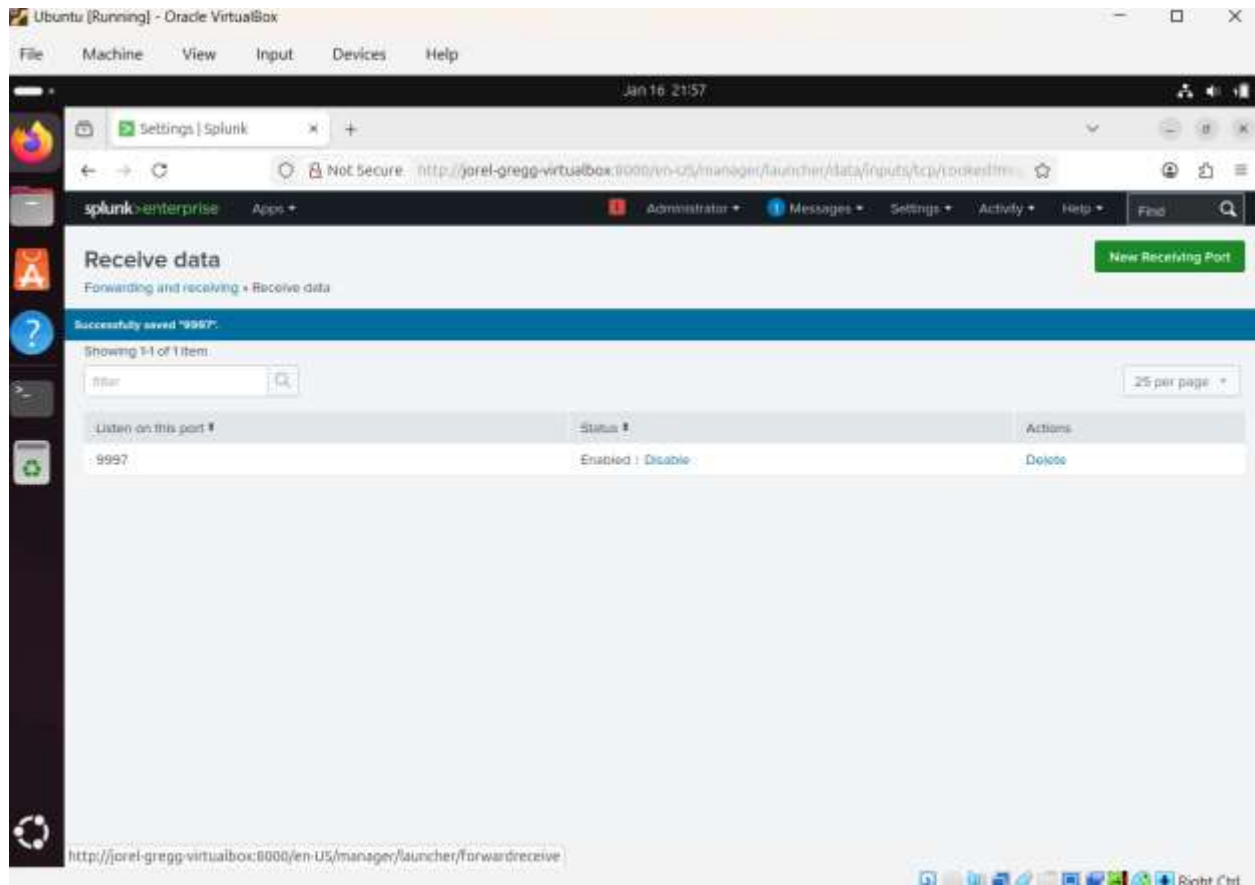
5. I then installed and ran the Splunk Forwarder program on my Kali-Linux virtual machine.



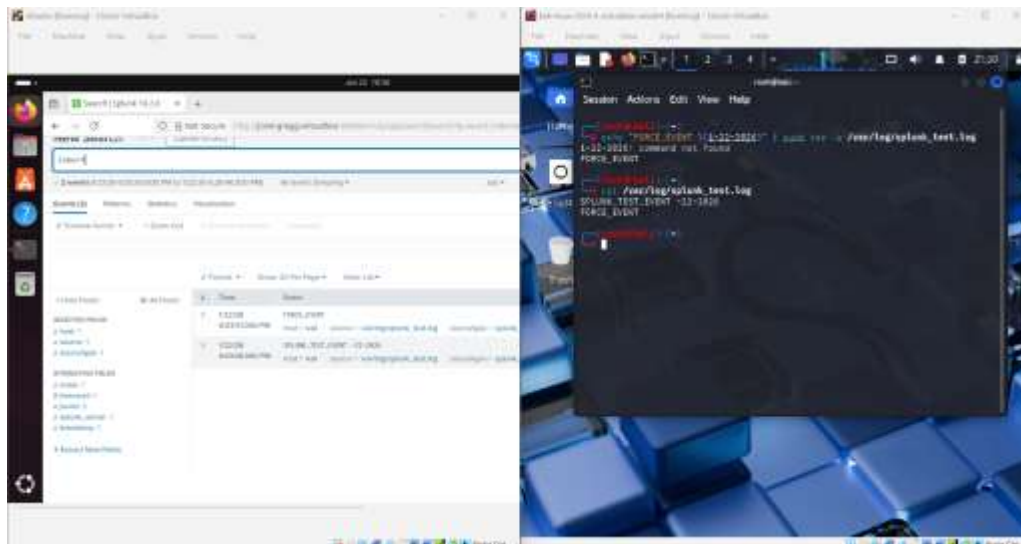
The screenshot shows a Kali Linux virtual machine interface. At the top, there is a menu bar with 'File', 'Machine', 'View', 'Input', 'Devices', and 'Help'. Below the menu bar is a toolbar with various icons and a status bar showing '0:55'. The main area displays a terminal window titled 'root@kali: ~'. The terminal output shows the Splunk Forwarder installation process, including checking the management port, creating directories, generating certificates, and starting the daemon.

```
root@kali: ~  
Session Actions Edit View Help  
Checking mgmt port [8089]: open  
Creating: /opt/splunkforwarder/var/lib/splunk  
Creating: /opt/splunkforwarder/var/run/splunk/appserver/i18n  
Creating: /opt/splunkforwarder/var/run/splunk/appserver/modules/static/css  
Creating: /opt/splunkforwarder/var/run/splunk/upload  
Creating: /opt/splunkforwarder/var/run/splunk/search_telemetry  
Creating: /opt/splunkforwarder/var/run/splunk/search_log  
Creating: /opt/splunkforwarder/var/spool/splunk  
Creating: /opt/splunkforwarder/var/spool/dirmoncache  
Creating: /opt/splunkforwarder/var/lib/splunk/authDb  
Creating: /opt/splunkforwarder/var/lib/splunk/hashDb  
Creating: /opt/splunkforwarder/var/run/splunk/collect  
Creating: /opt/splunkforwarder/var/run/splunk/sessions  
New certs have been generated in '/opt/splunkforwarder/etc/auth'.  
New certs have been generated in '/opt/splunkforwarder/etc/auth'.  
Checking conf files for problems ...  
Done  
Checking default conf files for edits ...  
Validating installed files against hashes from '/opt/splunkforwarder/splunkforwarder-10.2.0-d749cb17ea65-linux-amd64-manifest'  
All installed files intact.  
Done  
All preliminary checks passed.  
Starting splunk server daemon (splunkd) ...  
Done
```

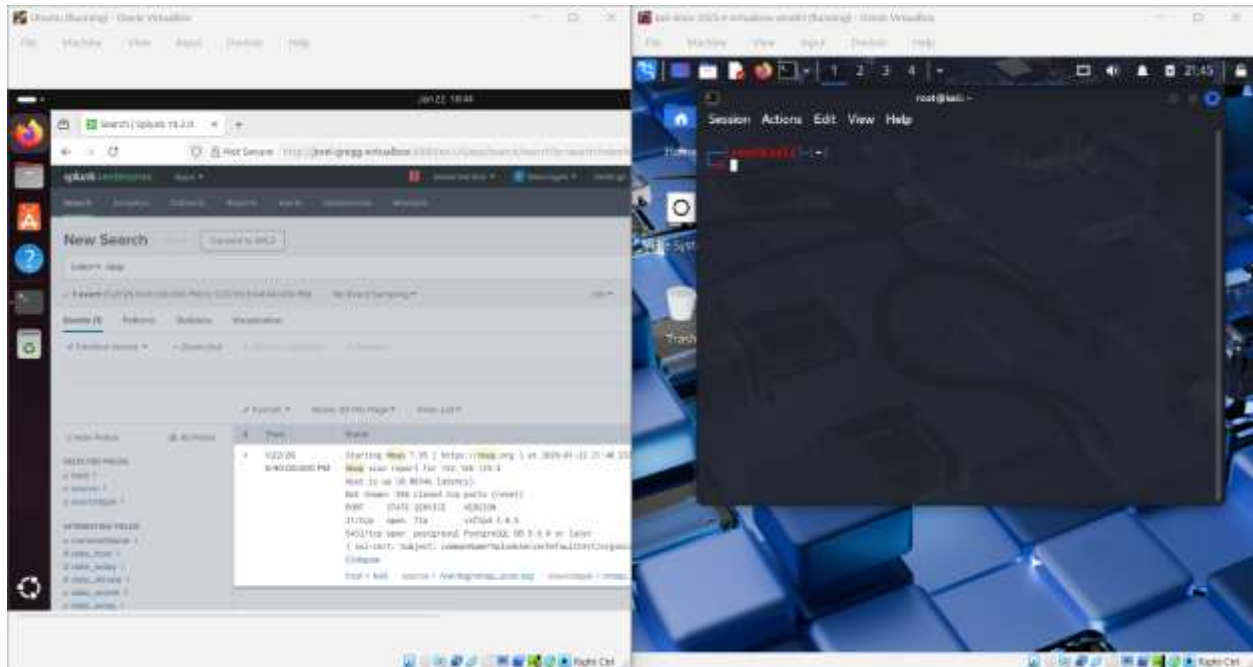

6. The next step was to ensure that I set up the Splunk receiving end on my Ubuntu machine to listen to port 9997.



7. Then I added my journal log in the Kali VM to my forwarder and restarted the forwarder. I couldn't find the journal log in the searches of Splunk so I had to set up a fake log file and create some events to ensure Splunk was receiving data.

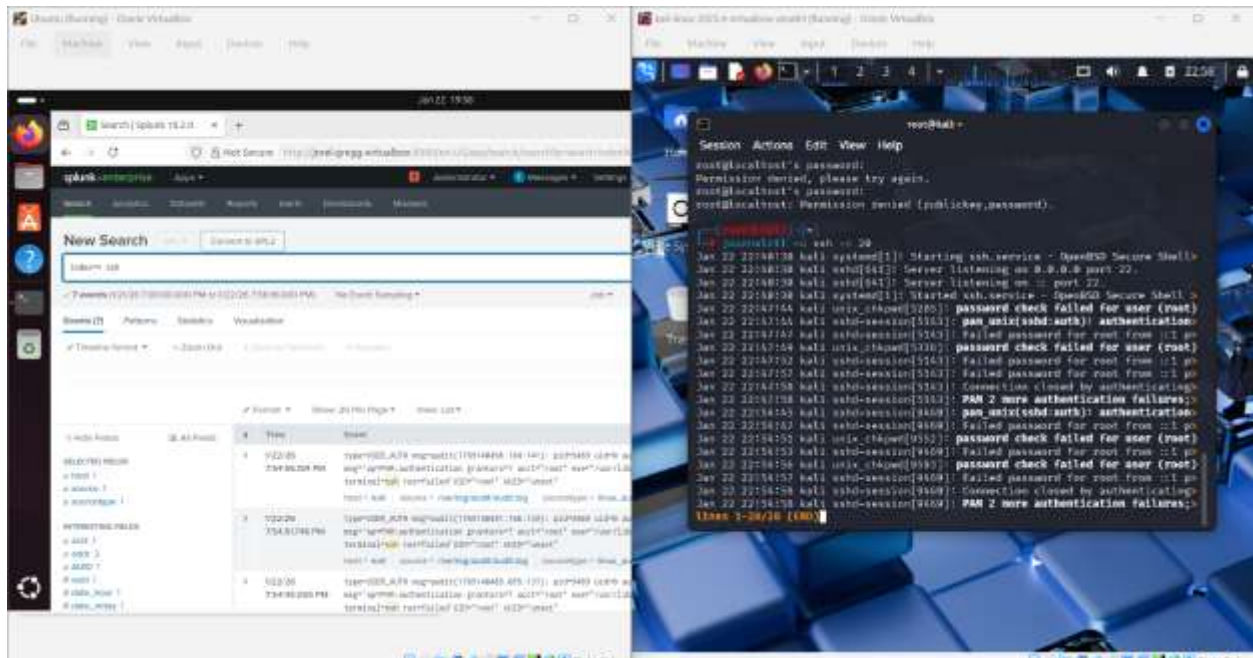


8. I began some with some basic threat actor actions like scanning with NMAP but I couldn't figure out why I couldn't find any logs on Splunk. After doing some research I discovered that NMAP is not really "logged" and had to create a log file and add it to the monitors. Then when I scanned, I saved it to the log file to generate logs to find.

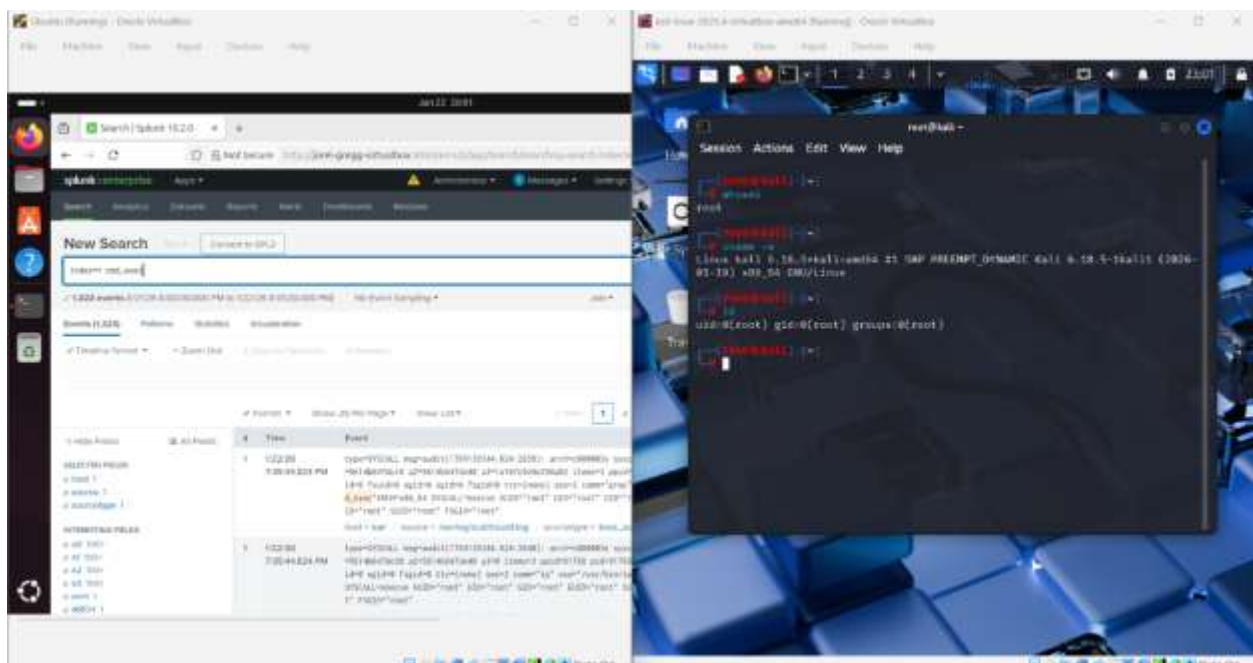


9. As I dug deeper into learning how SPLUNK works and finding what logs are NOT created in Kali, I researched and found that common commands needed to be set up in logs by an enterprise tool Audtid. I downloaded this onto Kali and configured it. While I was connected to the internet with my Kali VM I went ahead and grabbed ssh server as well so I could practice identifying that log data. I then turned the second network adapter off, so the machines were only on my local host network again.

10. First thing I attempted was to capture failed password attempts through SSH by just using ssh localhost and then the incorrect password three times. Then I went to my Splunk search and report and searched for index=* ssh which yielded results.



11. Last, I wanted to ensure that my Auditd was functioning, so I ran some simple commands such as whoami and id, then searched Splunk for index=* cmd_exec which showed that it was receiving the logs for the Auditd program.



This lab not only continued my education in learning Linux but allowed me to begin getting used to running the SIEM tool Splunk and identifying how to find log information that could potentially be a threat actor. My next goal is to continue learning what's capable with Splunk and begin setting up scheduled alerts based on the searches I completed in this lab.