Advanced Dynamic Nessus Scan Analyzed with Splunk

Naima Mumin

Purpose of this Project:

- In this project, my goal is to use both Nessus Scan and Splunk tools. First, I will conduct an advanced Dynamic Nessus Scan. After that, I will analyze these scans in Splunk, using the CSV scan data obtained from Nessus for further examination.

Understanding Nessus and Splunk:

- **Nessus:** is a vulnerability scanner designed to detect security vulnerabilities within the network.
- **Splunk:** is a Seim(Security Information and Event Management) tool, used for log analysis and network monitoring.

Operating Systems:

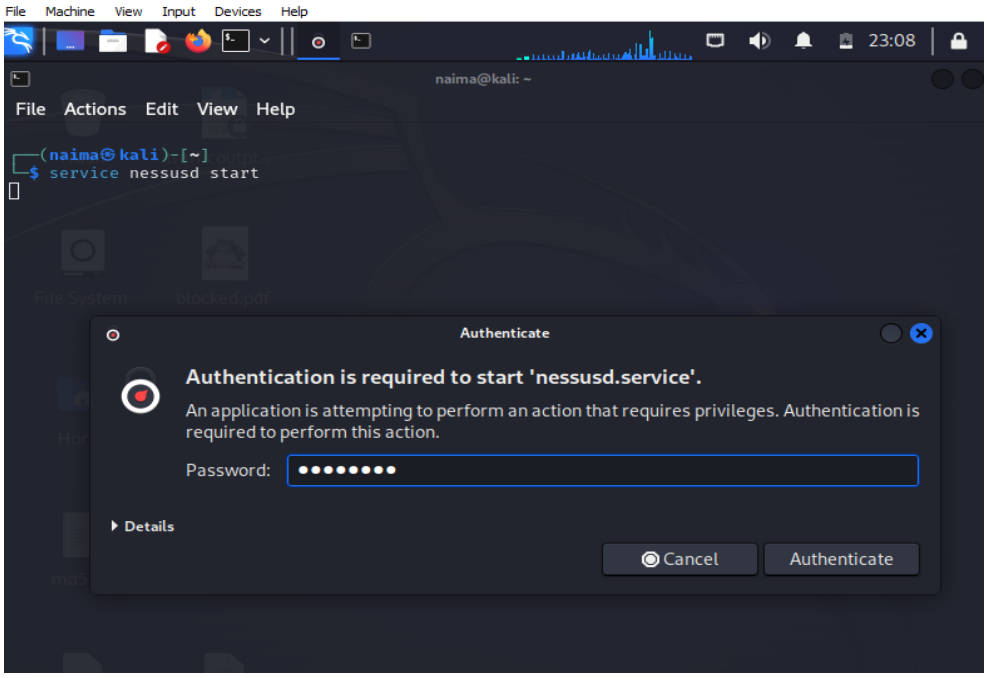
- **Kali Linux:** (Using it for the Advanced Nessus vulnerability scan)
- **Ubuntu:** (Hosts Splunk, enabling comprehensive analysis of the Nessus scan outcomes)

1. Start Nessus

➔ I open my kali Linux.

➔ Next, I enter the command “Service nessusd start” in the terminal to start the Nessus tool.

➔ When prompted, I input my Kali Linux password.



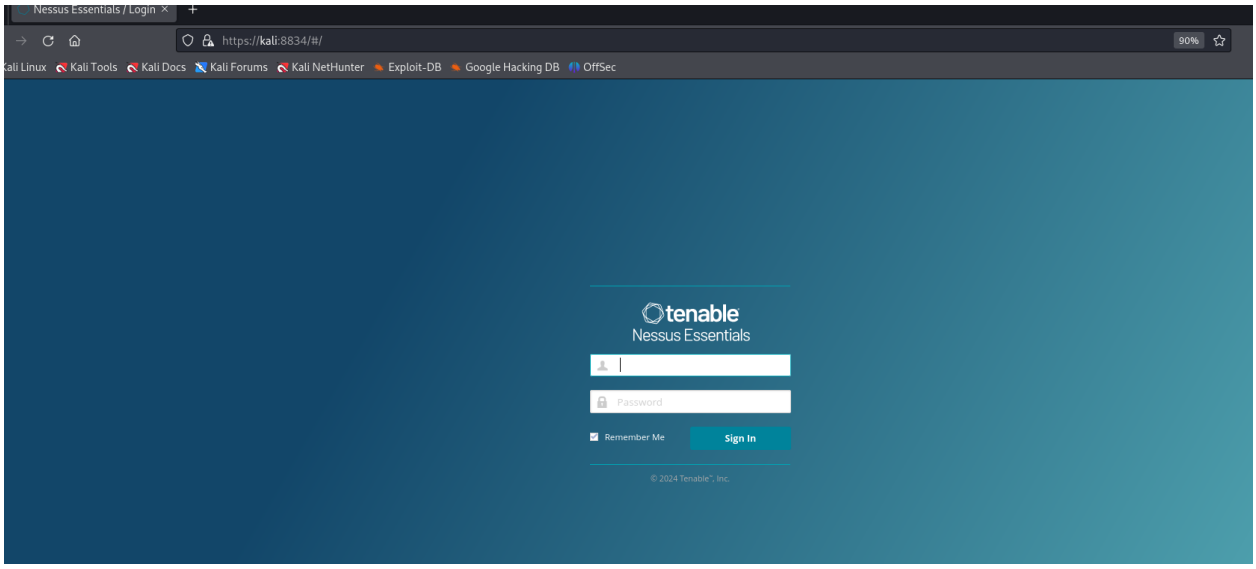
2. Verifying Nessus Status

➔ I check if Nessus is active and running using the command “systemctl status nessusd.”

- As seen in the image below, Nessus is confirmed as active and running. So now I can move on and start my scanning.

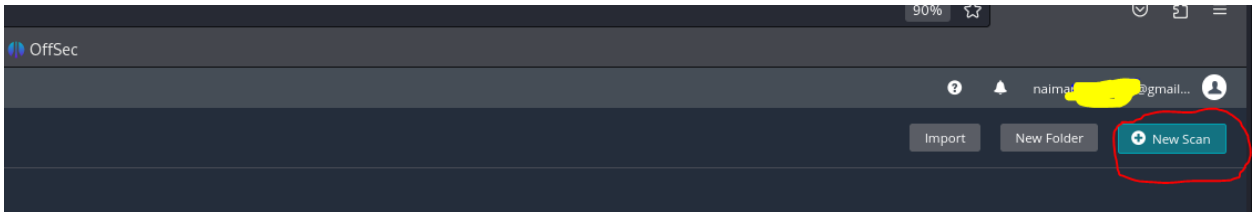
```
┌──(naima㉿kali)-[~]
└─$ systemctl status  nessusd
● nessusd.service - The Nessus Vulnerability Scanner
     Loaded: loaded (/lib/systemd/system/nessusd.service; enabled; preset: disabled)
     Active: active (running) since Tue 2024-01-02 23:08:32 CST; 4s ago
   Main PID: 3622 (nessus-service)
      Tasks: 17 (limit: 6821)
     Memory: 165.5M
        CPU: 5.765s
     CGroup: /system.slice/nessusd.service
             ├─3622 /opt/nessus/sbin/nessus-service -q
             └─3626 nessusd -q

Jan 02 23:08:32 kali systemd[1]: Started nessusd.service - The Nessus Vulnerability Scanner.

┌──(naima㉿kali)-[~]
```

3. Go to the Nessus website and start putting my email and password.



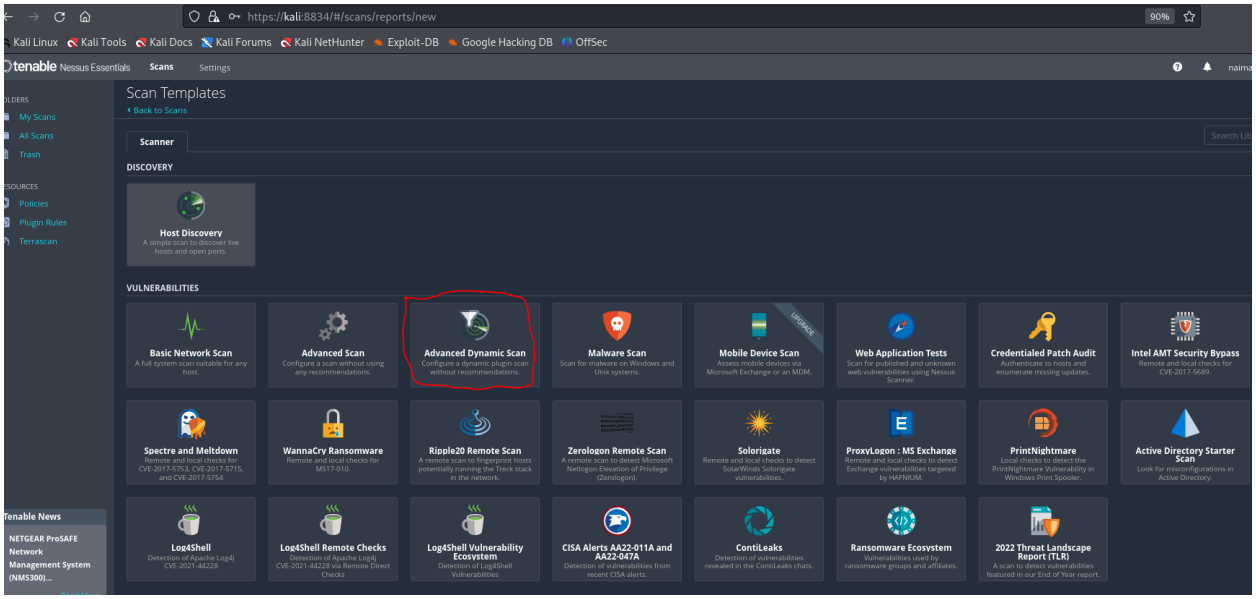
4. Start Scanning

➔ To start scanning, I will simply click the “New Scan” button in the top right corner and choose the type of scan I need.



5. Determining My Scan Preference

➔ In this Project, I'll use the "Advance Dynamic Scan" to deeply explore my network's vulnerabilities and thoroughly evaluate software risks. This scan's extensive analysis sets it apart from other options in Nessus.



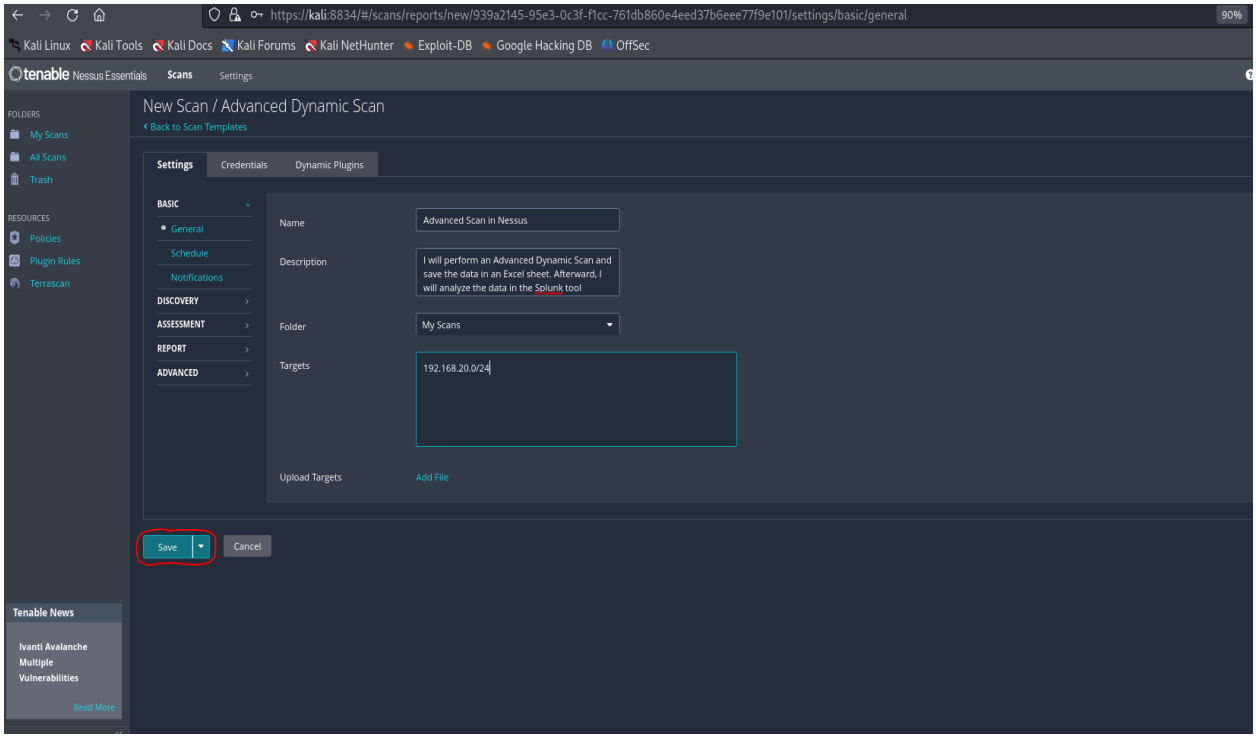
6. Adding the information I need in my scan

➔ The name of this scan will be "Advanced Scan in Nessus".

➔ I added a brief description, as can be seen below to clarify the purpose of my Nessus scan.

➔ The folder I am going to save my scan is called "My Scans."

➔ And finally, I decided to scan all the IP addresses on my networking using "192.168.20.0/24

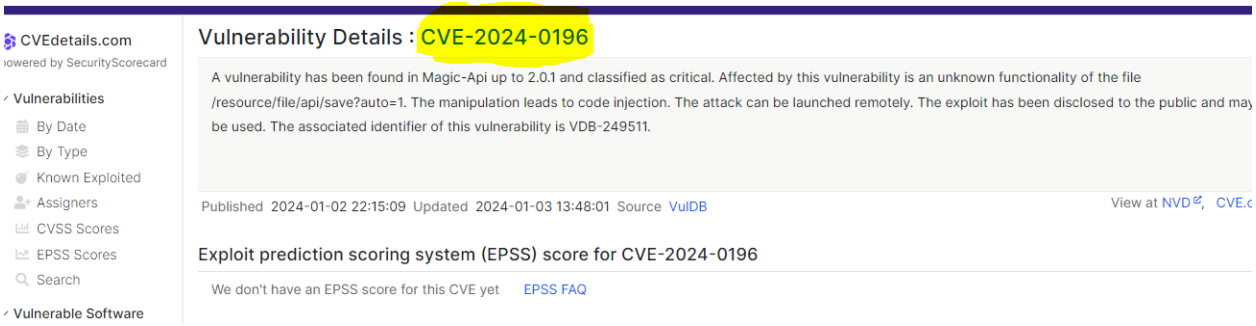


7. Choose The CVE

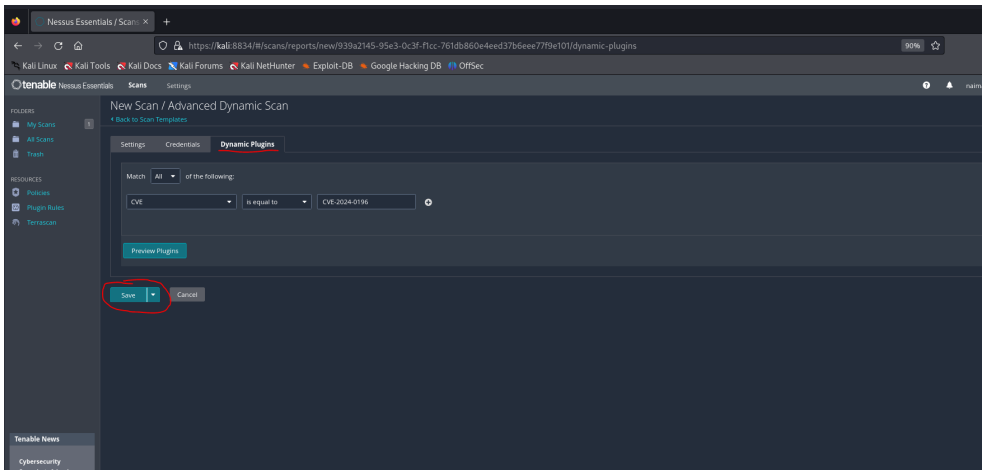
What is CVE?

- ➔ It stands for Common Vulnerability and Exposure. It is a unique code for a recognized security issue in software or hardware. I've included CVE-2024-0196 in my Nessus scan to specifically check for and address this known vulnerability within my network. This helps me to ensure that my systems are secure by identifying and fixing any issues highlighted by this code.

❖ Website I got the CVE I'm using in my Nessus https://www.cvedetails.com/cve/CVE-2024-0196/ . In this website, there are lots of CVE in there, but I'm interested using this CVE-2024-0196

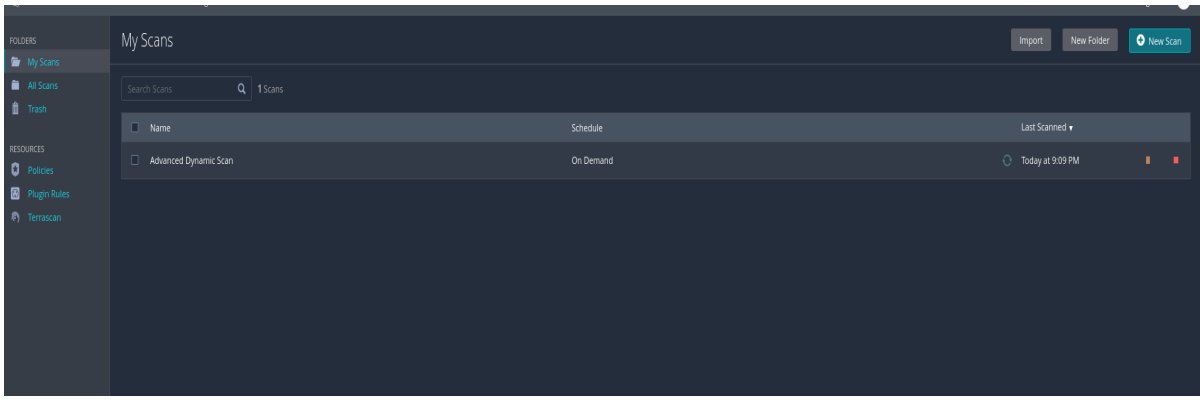


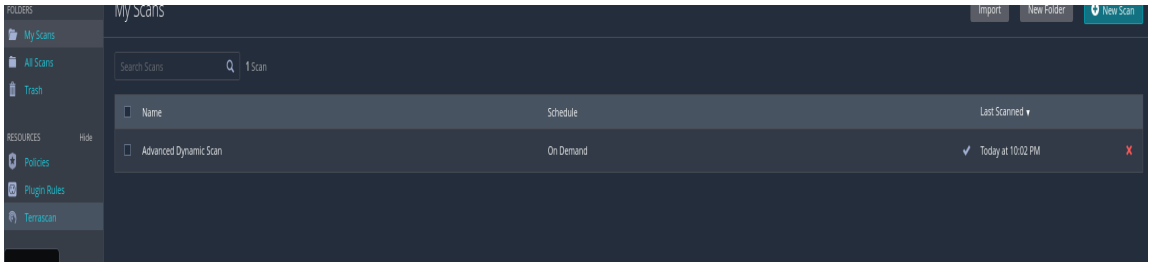
- Once I've chosen the CVE, the next step is to "Save' my scan.



8. Check if it is launching

- ➔ The “last Scanned” section displays two arrows circling each other, indicating that the scan is in progress. Once a checkmark shows up, it signifies the scan’s completion, allowing me to access and review the results of the scan.





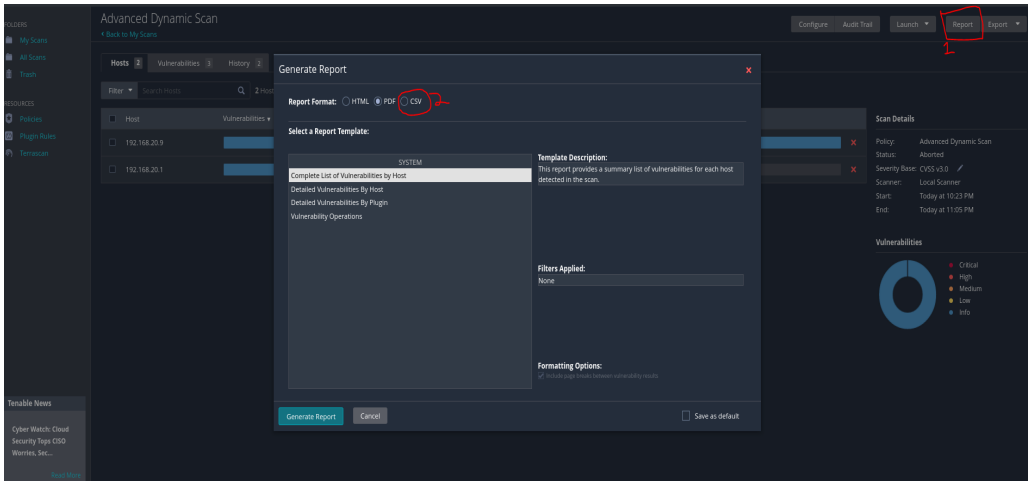
9. Save the scan as a CSV

What is CSV?

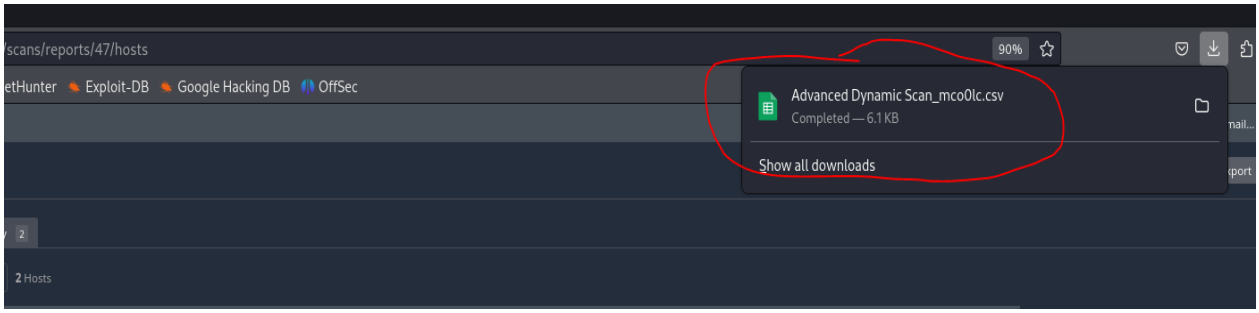
- ➔ It’s similar to a structured report card summarizing the scan, allowing quick identification of vulnerabilities and affected systems.

Here is what I’m going to do to save this scan as a CSV file:

- ➔ Click on “Report” in the right upper corner.
- ➔ Select “CSV” as the report format.
- ➔ Complete the process by clicking on “Generate Report”.



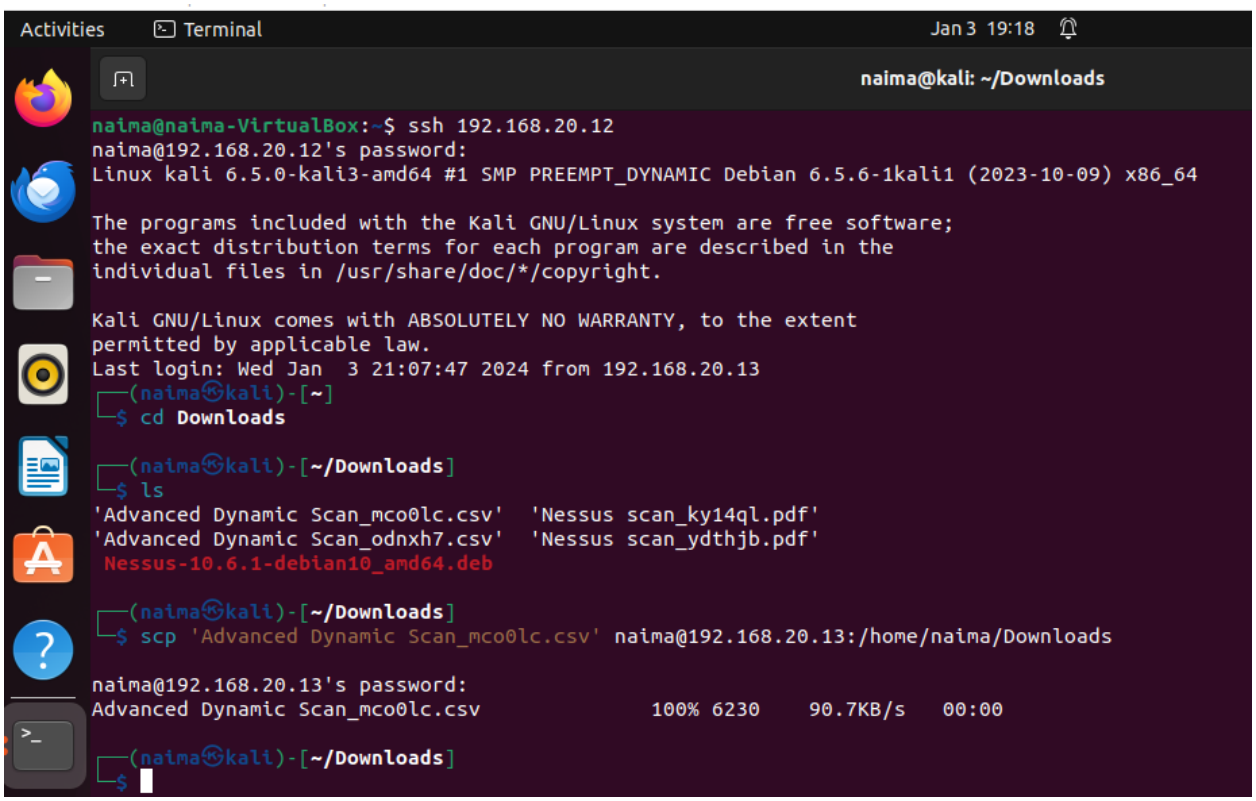
- Now the scan is being saved as a CSV file.



10. Use scp to copy the file from Kali Linux to Ubuntu

➔ First I logged in to my Ubuntu machine and I'm going to start the SSH.

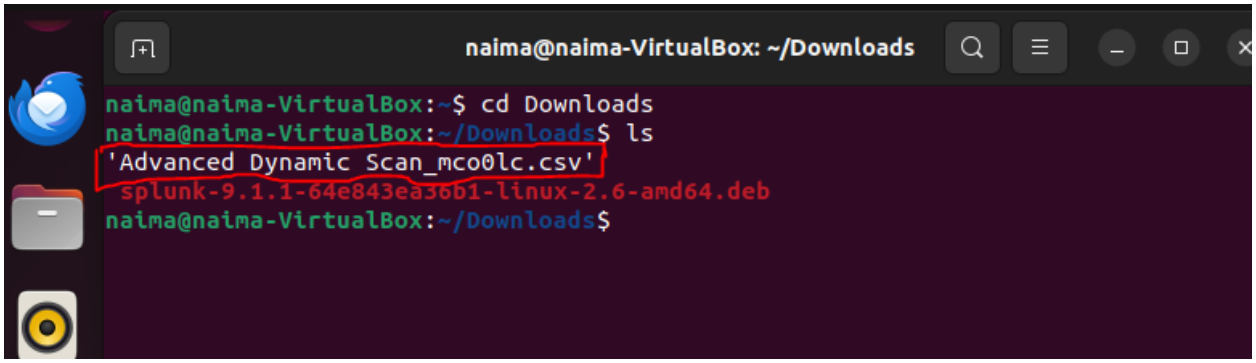
- ➔ Connected to Kali Linux using “SSH 192.168.20.12” and provided the password.
- ➔ Navigated to the “Downloads” folder using “cd Downloads”.
- ➔ Used “scp” and it stands for Secure Copy Protocol. This helps me to copy files securely between my Kali Linux Machine to Ubuntu.
- ➔ Executed the command: **scp ‘Advanced Dynamic Scan_mco0lc.csv’ naima@192.168.20.13:/home/naima/Dowloads**



11. Check if I correctly download the file in my Ubuntu

- ➔ open a new terminal on my Ubuntu machine.
- ➔ Enter the command: **cd Downloads** and then **ls**

The image below confirms the successful transfer of the file from my Linux machine to my Ubuntu Machine.

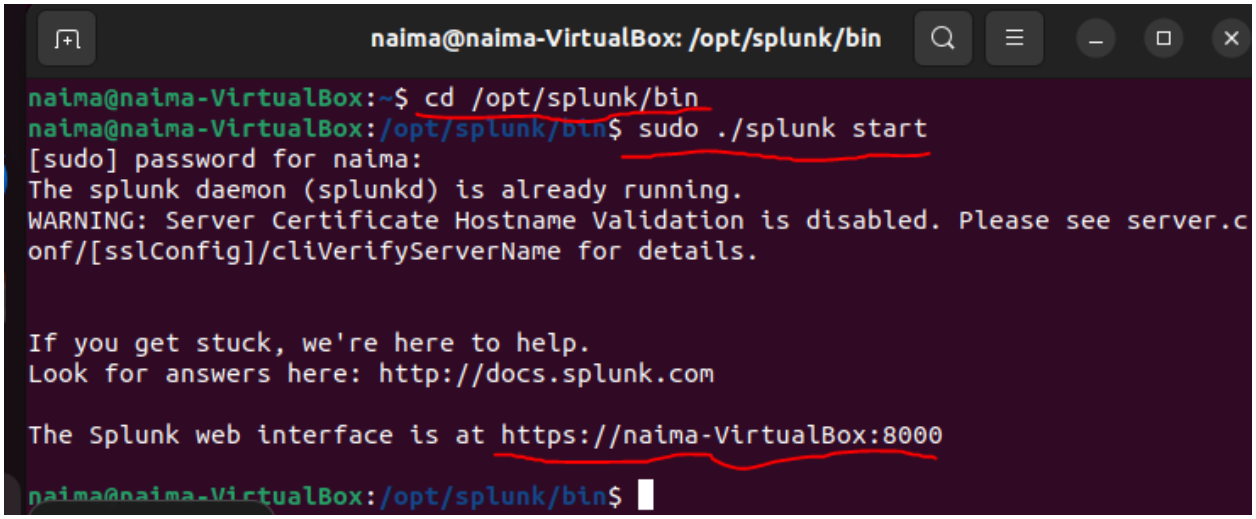


12. Start Splunk for Analyzing

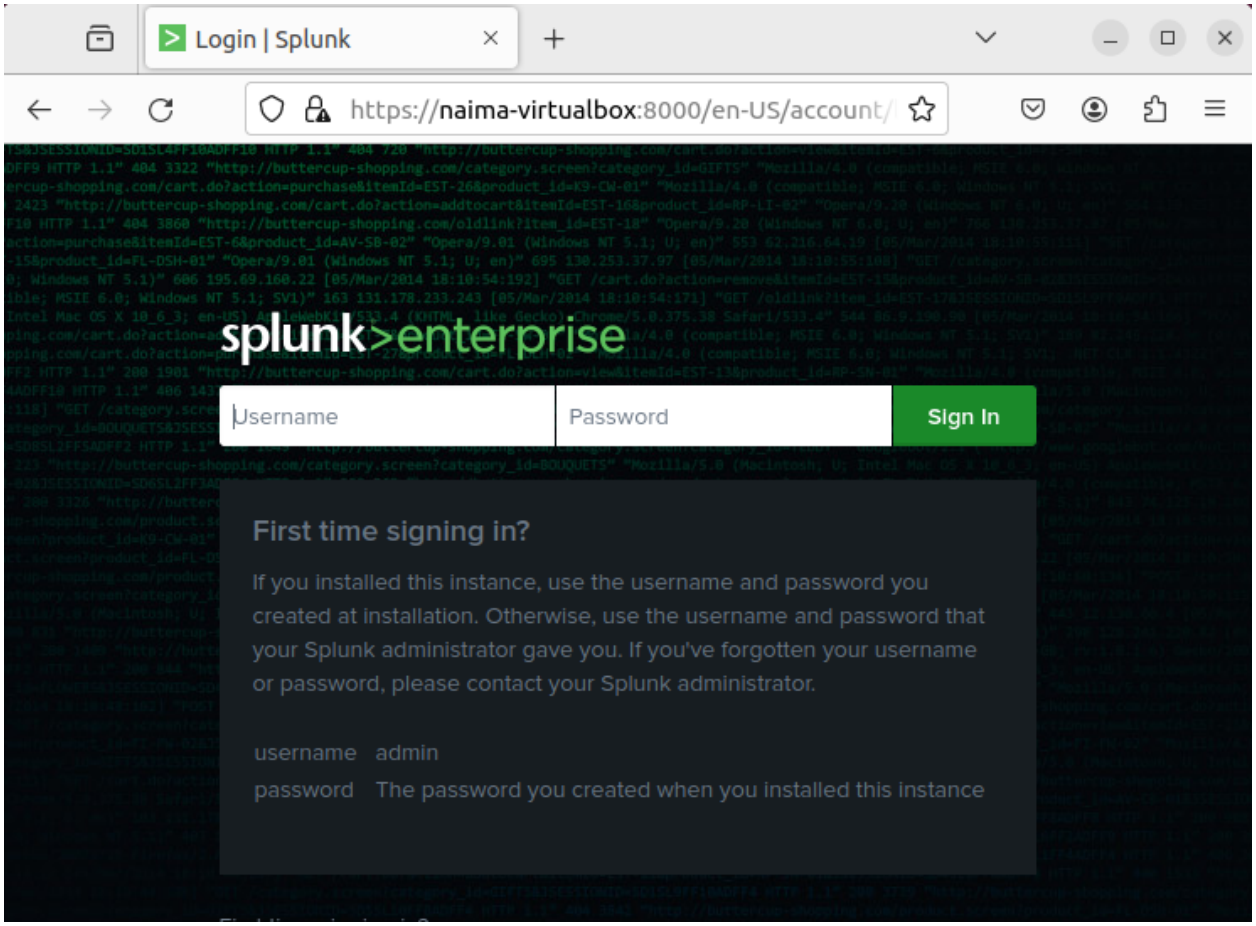
To initiate Splunk for analysis within my project, I'm executing the following commands:

- **Navigating to Splunk Bin Folder:**
  - ➔ I navigate to **cd /opt/splunk/bin ,** the location where Splunk's executable files are stored.
- **Initiating Splunk with Sudo:**
  - ➔ Using **sudo ./splunk start** , I start Splunk with administrative privileges for analysis.
- **Password Confirmation:**
  - ➔ Upon starting Splunk, I confirm access by entering my Password as required.

Now, I'll copy the "https://naima-VirtualBox:8000" and paste it into the search bar of Firefox.

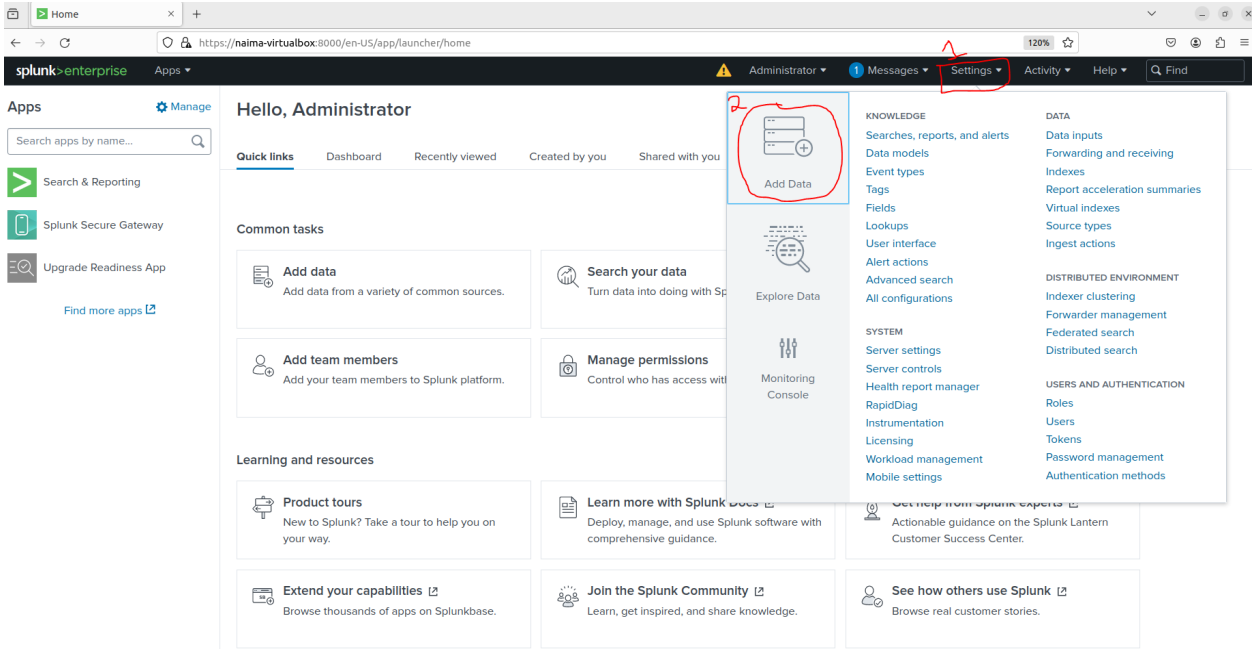


- Here, I'm going to log in Splunk with my username and password.

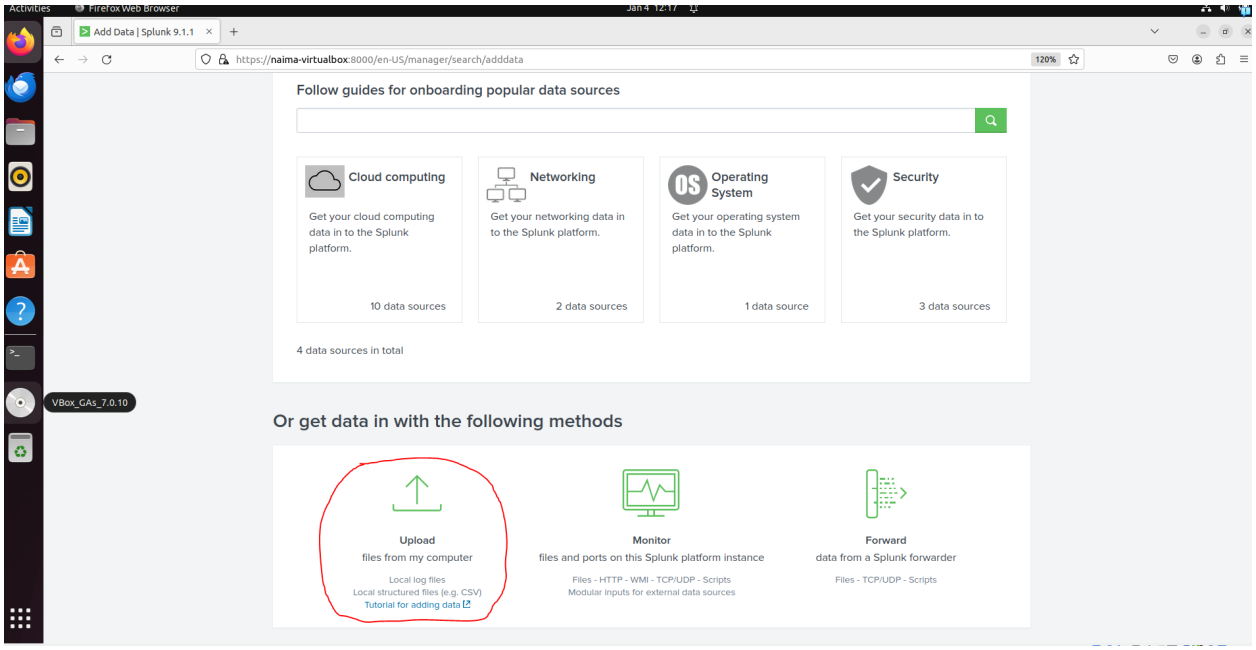


13. Upload the CSV file in Splunk

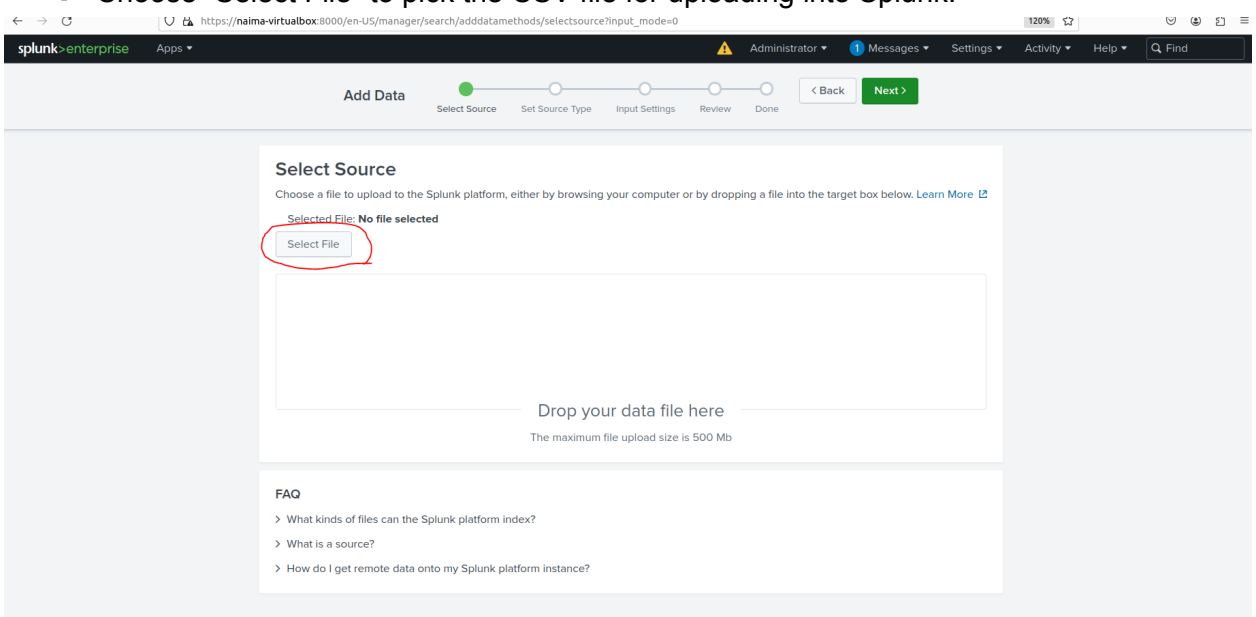
➔ Navigate to the “Settings” tab and select “Add Data” to initiate the CSV file upload process into Splunk.



➔ Then I'm going to click on where it says "Upload" to proceed.

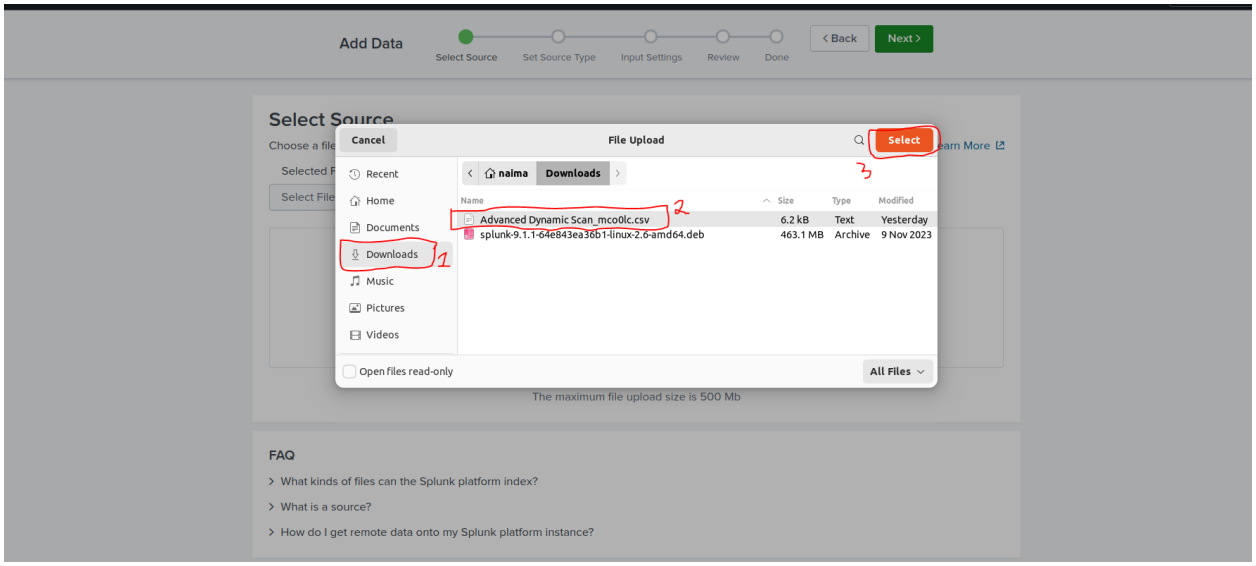


➔ Choose "Select File" to pick the CSV file for uploading into Splunk.

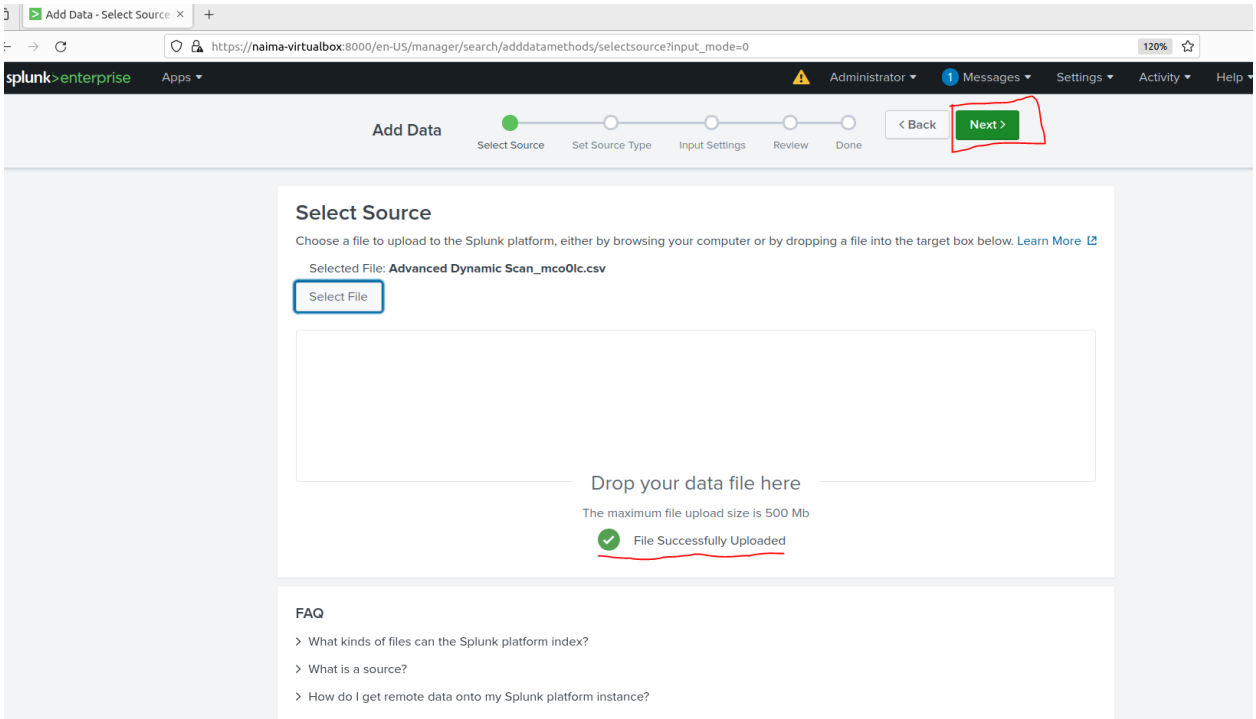


➔ After that, I accessed my "Downloads" folder on Ubuntu and opened the file named 'Advanced Dynamic Scan_mco0lc.csv' that I had created.

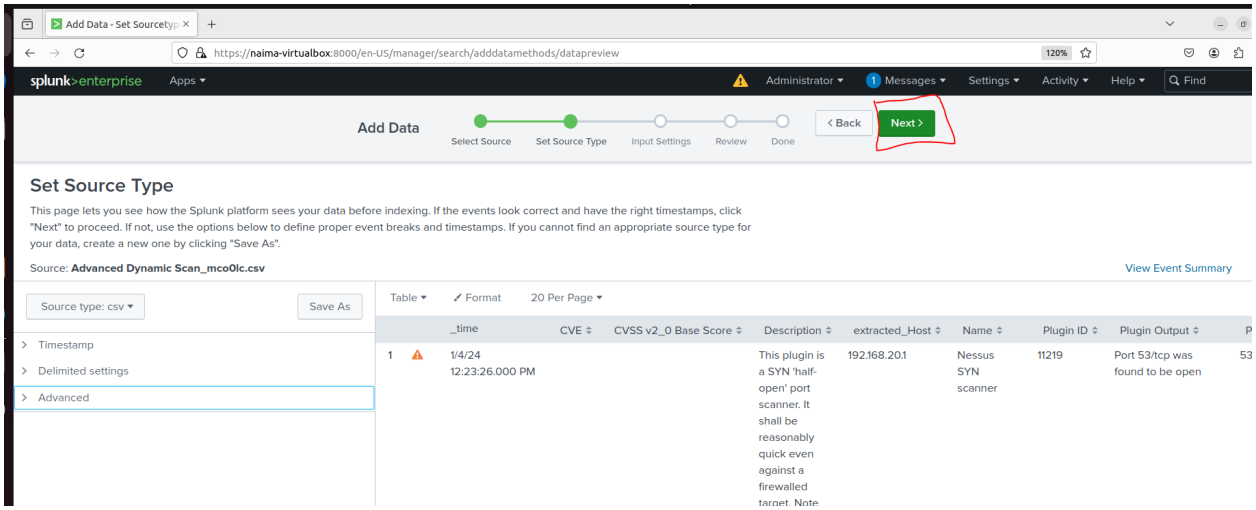
➔ Then I clicked on the "Select" button.



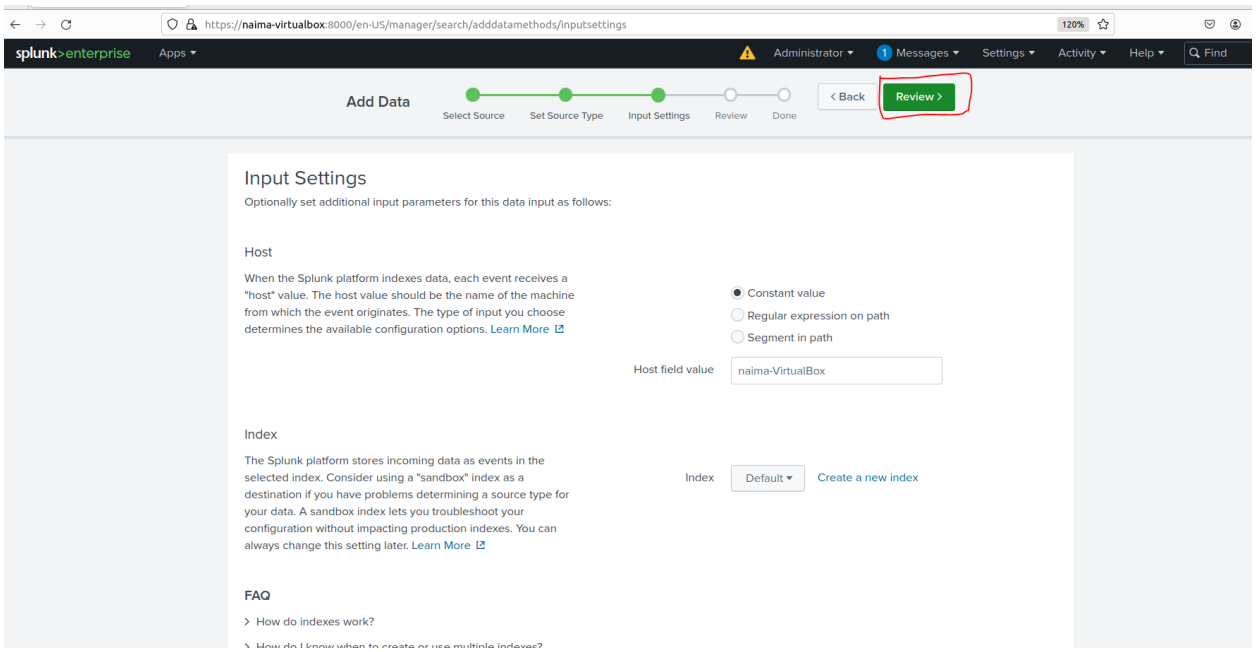
➔ I have successfully uploaded the file into Splunk. To proceed, I clicked on the "Next" button.



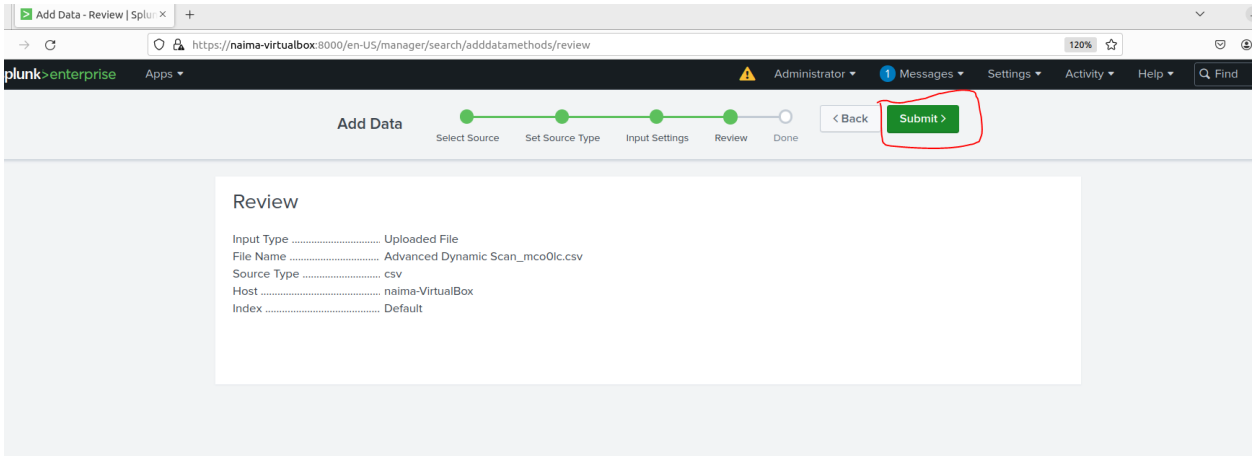
➔ Then click "Next".



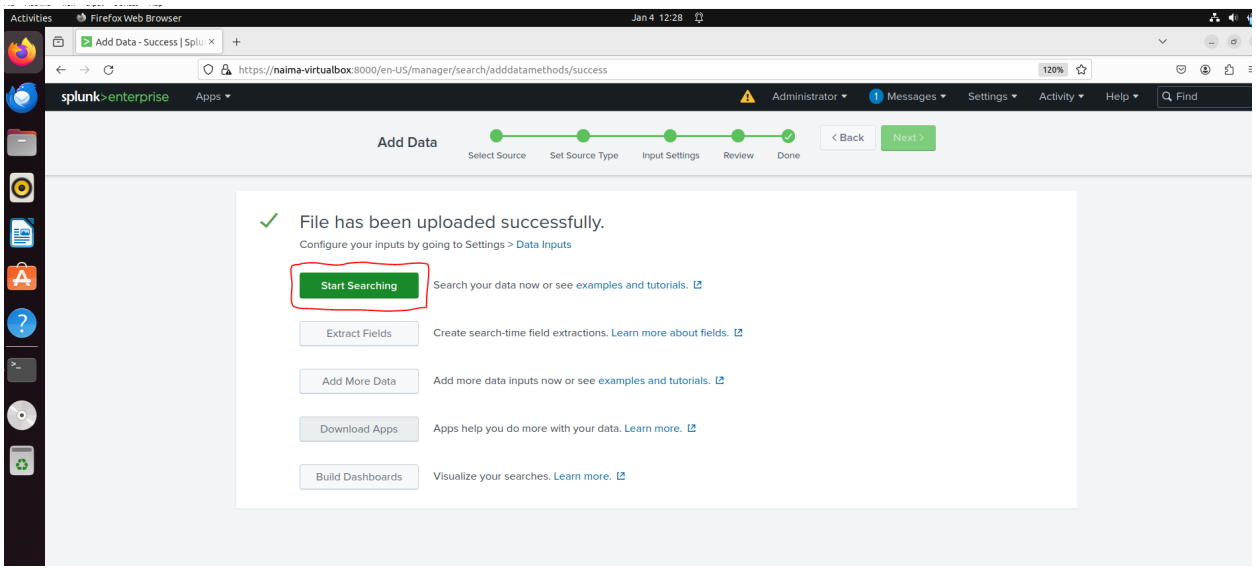
➔ To continue, I'm clicking on the "Review" button.



➔ Confirming that all my data are accurate, I'll proceed by clicking both the "Submit" and then "Start Searching" buttons.

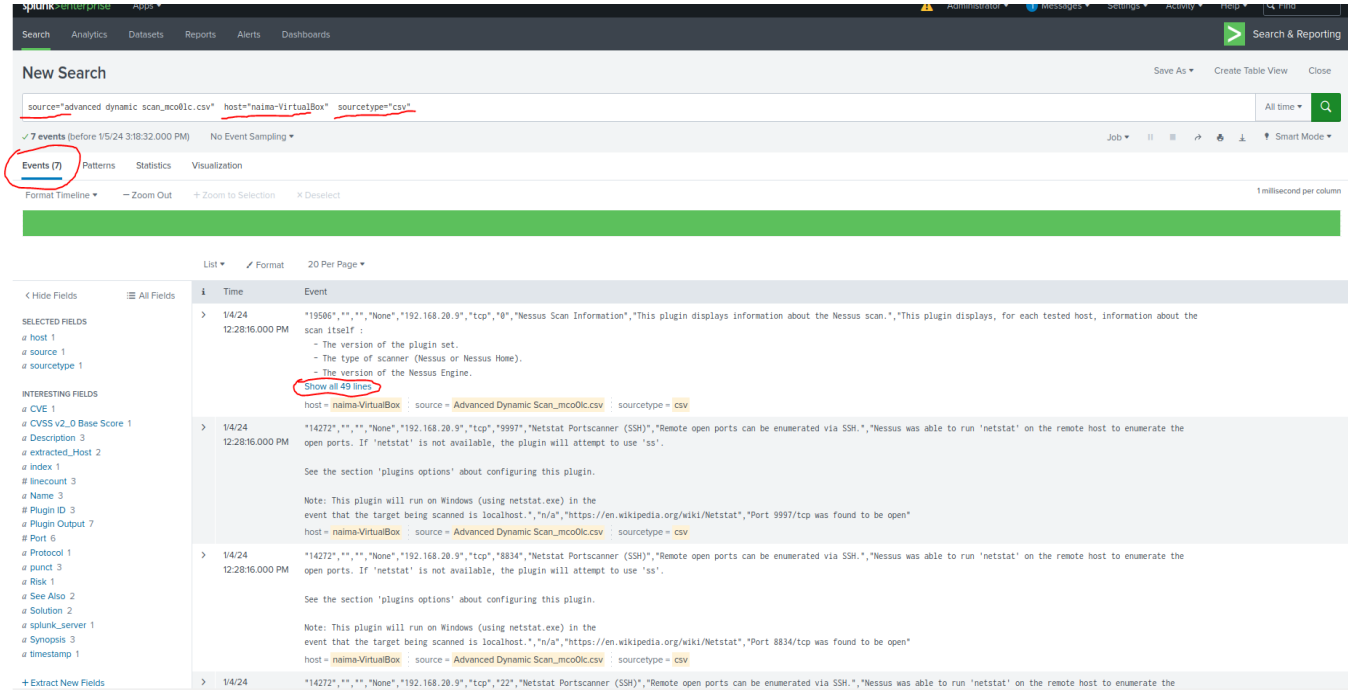


➔



14. Begin analyzing the Uploaded scan file in Splunk

➔ In the Search bar, I've got information on the source, host, and sourcetype for analysis in Splunk. Currently, there are 7 events associated with this scan. I'm going to focus on the first event now. Clicking on "Show all 49 lines" will expand the data, giving me a clearer view for analysis.



- ➔ For the first images below, they present details about a Nessus scan. This information covers things like the tools used for the scan, what was checked, and various technical details. It tells us about the version of Nessus used, the settings applied during the scan, and specifics like the scan's start time and duration. Essentially, it's a summary that gives us a clear picture of how the scan was conducted and what it looked for on the network.

Why is this important?

- ➔ Because it is crucial for assessing and improving network security.

i	Time	Event
>	1/4/24 12:28:16.000 PM	"19506","","","None","192.168.20.9","tcp","0","Nessus Scan Information","This plugin displays information about the Nessus scan.","This plugin displays, for each tested host, information scan itself : - The version of the plugin set. - The type of scanner (Nessus or Nessus Home). - The version of the Nessus Engine. - The port scanner(s) used. - The port range scanned. - The ping round trip time - Whether credentialed or third-party patch management checks are possible. - Whether the display of superseded patches is enabled - The date of the scan. - The duration of the scan. - The number of hosts scanned in parallel. - The number of checks done in parallel.","n/a","","Information about this scan : Nessus version : 10.6.4 Nessus build : 20005 Plugin feed version : 202401021845 Scanner edition used : Nessus Home Scanner OS : LINUX Scanner distribution : debian10-x86-64 Scan type : Normal Scan name : Advanced Dynamic Scan Scan policy used : Advanced Dynamic Scan Scanner IP : 192.168.20.9 Ping RTT : Unavailable Thorough tests : no Experimental tests : no Plugin debugging enabled : no Paranoia level : 1 Report verbosity : 1 Safe checks : yes Optimize the test : yes Credentialed checks : no Patch management checks : None Display superseded patches : yes (supersedence plugin launched)

```
CGI scanning : disabled
Web application tests : disabled
Max hosts : 30
Max checks : 5
Recv timeout : 5
Backports : None
Allow post-scan editing : Yes
Nessus Plugin Signature Checking : Enabled
Audit File Signature Checking : Disabled
Scan Start Date : 2024/1/2 22:23 CST
Scan duration : 9 sec
Scan for malware : no
"
```

Collapse

host = naima-VirtualBox | source = Advanced Dynamic Scan_mco0lc.csv | sourcetype = csv

➔ From the visuals displayed below, I observed details regarding two hosts: 192.168.20.9 and 192.168.20.1. Specifically, 192.168.20.9 displayed open ports such as: 9997(used by Splunk for checking data), 8837(Not commonly used), and 22(for secure connections), while 192.168.20.1 showed open ports for 15000(similar to 8837) and 53(for DNS). These findings suggest potential vulnerabilities or accessible services on each host due to the open ports detected.

INTERESTING FIELDS

- # CVE 1
- # CVSS v2_0 Base Score 1
- # Description 3
- # extracted_Host 2
- # index 1
- # linecount 3
- # Name 3
- # Plugin ID 3
- # Plugin Output 7
- # Port 6
- # Protocol 1
- # punct 3
- # Risk 1
- # See Also 2
- # Solution 2
- # splunk_server 1
- # Synopsis 3
- # timestamp 1

+ Extract New Fields

i	Time	Event
>	1/4/24 12:28:16.000 PM	"14272","","","None","192.168.20.9","tcp","9997","Netstat Portscanner (SSH)","Remote open ports can be enumerated via SSH.","Nessus was able to run 'netstat' on the remote host to enumerate the open ports. If 'netstat' is not available, the plugin will attempt to use 'ss'. See the section 'plugins options' about configuring this plugin. Note: This plugin will run on Windows (using netstat.exe) in the event that the target being scanned is localhost.","n/a","https://en.wikipedia.org/wiki/Netstat","Port 9997/tcp was found to be open" host = naima-VirtualBox ; source = Advanced Dynamic Scan_mco0lc.csv ; sourcetype = csv
>	1/4/24 12:28:16.000 PM	"14272","","","None","192.168.20.9","tcp","8834","Netstat Portscanner (SSH)","Remote open ports can be enumerated via SSH.","Nessus was able to run 'netstat' on the remote host to enumerate the open ports. If 'netstat' is not available, the plugin will attempt to use 'ss'. See the section 'plugins options' about configuring this plugin. Note: This plugin will run on Windows (using netstat.exe) in the event that the target being scanned is localhost.","n/a","https://en.wikipedia.org/wiki/Netstat","Port 8834/tcp was found to be open" host = naima-VirtualBox ; source = Advanced Dynamic Scan_mco0lc.csv ; sourcetype = csv
>	1/4/24 12:28:16.000 PM	"14272","","","None","192.168.20.9","tcp","22","Netstat Portscanner (SSH)","Remote open ports can be enumerated via SSH.","Nessus was able to run 'netstat' on the remote host to enumerate the open ports. If 'netstat' is not available, the plugin will attempt to use 'ss'. See the section 'plugins options' about configuring this plugin. Note: This plugin will run on Windows (using netstat.exe) in the event that the target being scanned is localhost.","n/a","https://en.wikipedia.org/wiki/Netstat","Port 22/tcp was found to be open" host = naima-VirtualBox ; source = Advanced Dynamic Scan_mco0lc.csv ; sourcetype = csv

>	1/4/24 12:28:16.000 PM	"11219","","","None","192.168.20.1","tcp","15000","Nessus SYN scanner","It is possible to determine which TCP ports are open.","This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target. Note that SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target, if the network is loaded.","Protect your target with an IP filter.","","Port 15000/tcp was found to be open" Collapse host = naima-VirtualBox ; source = Advanced Dynamic Scan_mco0lc.csv ; sourcetype = csv
>	1/4/24 12:28:16.000 PM	"11219","","","None","192.168.20.1","tcp","53","Nessus SYN scanner","It is possible to determine which TCP ports are open.","This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target. Note that SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target, if the network is loaded.","Protect your target with an IP filter.","","Port 53/tcp was found to be open" Collapse host = naima-VirtualBox ; source = Advanced Dynamic Scan_mco0lc.csv ; sourcetype = csv

Conclusion:

➔ In this project, I used Nessus for a deep vulnerability scan and analyzed the results in Splunk. Running the Advanced Dynamic Nessus Scan taught me how to identify and address security issues in a network. It highlighted the importance of regular checks for maintaining a secure network.