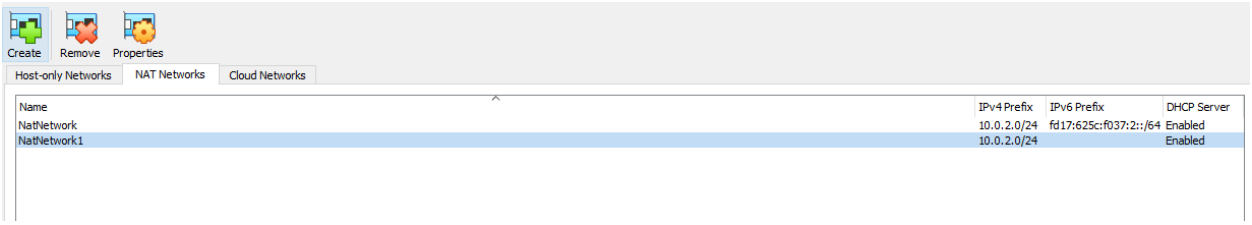
**Task 1:**

A company should invest their security resources towards ensuring that credentials can't be compromised. According to the article, it says that 80% of cyberattacks used legitimate credentials to evade detection. This means that they were able to steal an employee's or customer's credentials to access private information while going undetected.

There has been a rise in attempts to exploit authentication services, making it important to use strong authentication methods with multiple layers of defense to protect customer credentials. A few examples of credentials being stolen are the Slippy Spider and Scattered Spider groups. These groups used different technologies to steal personal customer data from big companies, and then they threatened to leak the information.

Hackers may target interactive services, which results in taking advantage of users' lack of awareness about how these systems work. For instance, they will use phishing techniques to trick users into revealing sensitive information. Also, cloud services have become a popular target for cyberattacks. Recent attacks often aim to steal the identity of vulnerable users. To counter this, it's important to implement rigorous identity verification measures to minimize the impact of new cloud-based cyberattack methods. It could also help to ensure that customers aren't using the same passwords. It is also important to ensure that our security stays up to date with all the new technology. This will ensure that it can keep up with the new methods that hackers will be using to exploit systems.

**Task 2:**



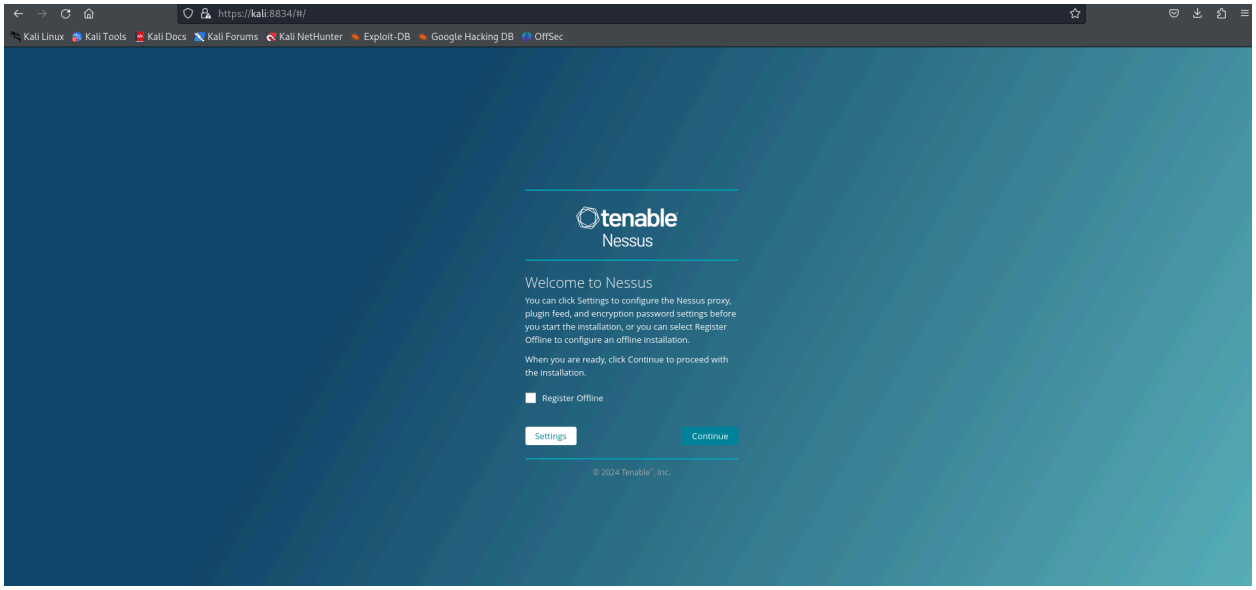

In this step, I created a new “NatNetork” and have set all the VM’s to the “Nat Network” setting.

```
┌──(timothyd㉿kali)-[~]
└─$ cd ~/Downloads

┌──(timothyd㉿kali)-[~/Downloads]
└─$ sudo dpkg -i Nessus*
Selecting previously unselected package nessus.
(Reading database ... 400164 files and directories currently installed.)
Preparing to unpack Nessus-10.7.1-debian10_amd64.deb ...
Unpacking nessus (10.7.1) ...
Setting up nessus (10.7.1) ...
HMAC : (Module_Integrity) : Pass
SHA1 : (KAT_Digest) : Pass
SHA2 : (KAT_Digest) : Pass
SHA3 : (KAT_Digest) : Pass
TDES : (KAT_Cipher) : Pass
AES_GCM : (KAT_Cipher) : Pass
AES_ECB_Decrypt : (KAT_Cipher) : Pass
RSA : (KAT_Signature) : RNG : (Continuous_RNG_Test) : Pass
Pass
```

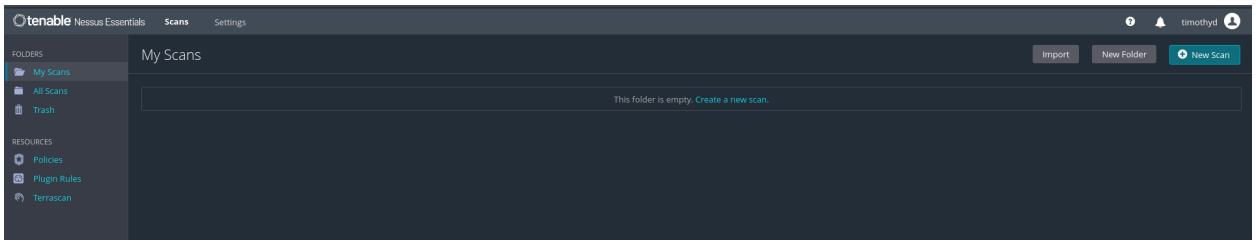
In this step, I am navigating to the downloads folder and installing the Nessus package.

```
┌──(timothyd㉿kali)-[~/Downloads]
└─$ sudo /bin/systemctl start nessusd.service

┌──(timothyd㉿kali)-[~/Downloads]
└─$
```

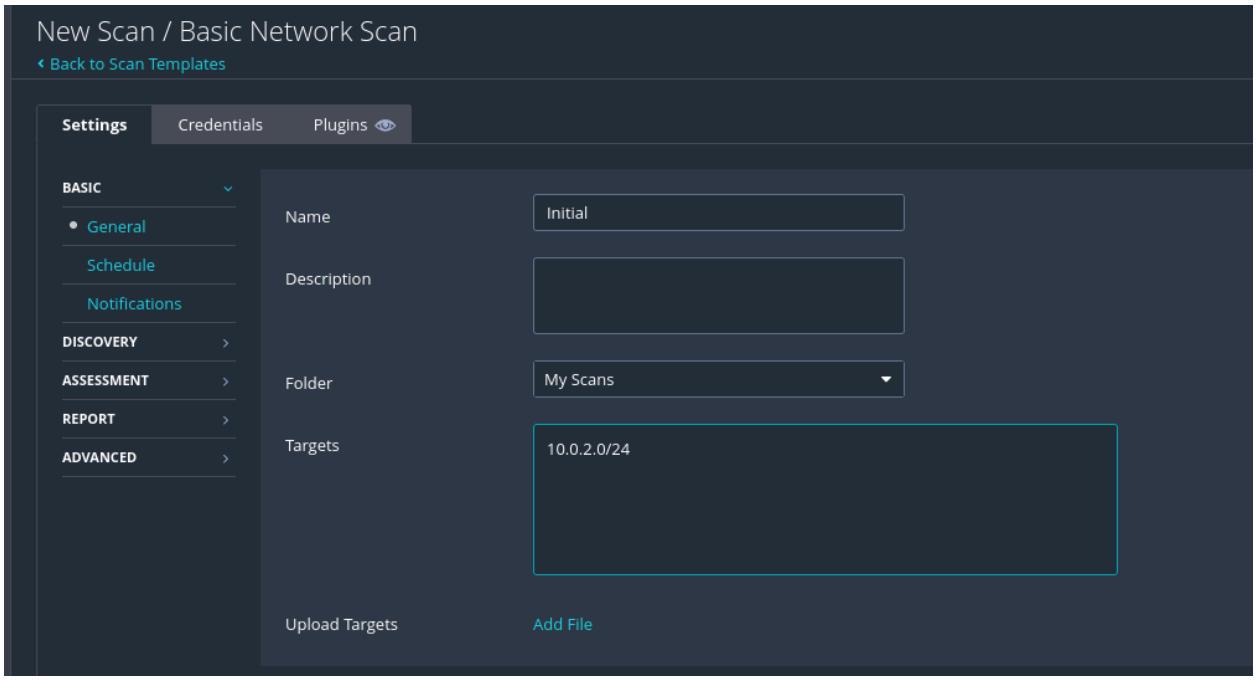
In this step, I am starting the Nessus daemon.



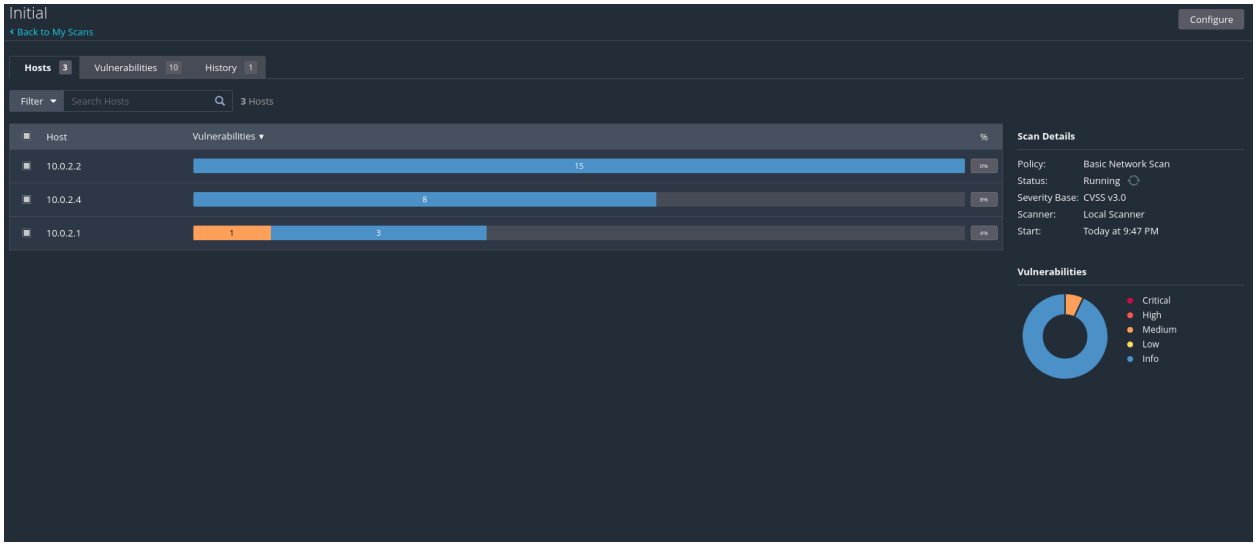

In this step, I have accessed the Nessus console locally.



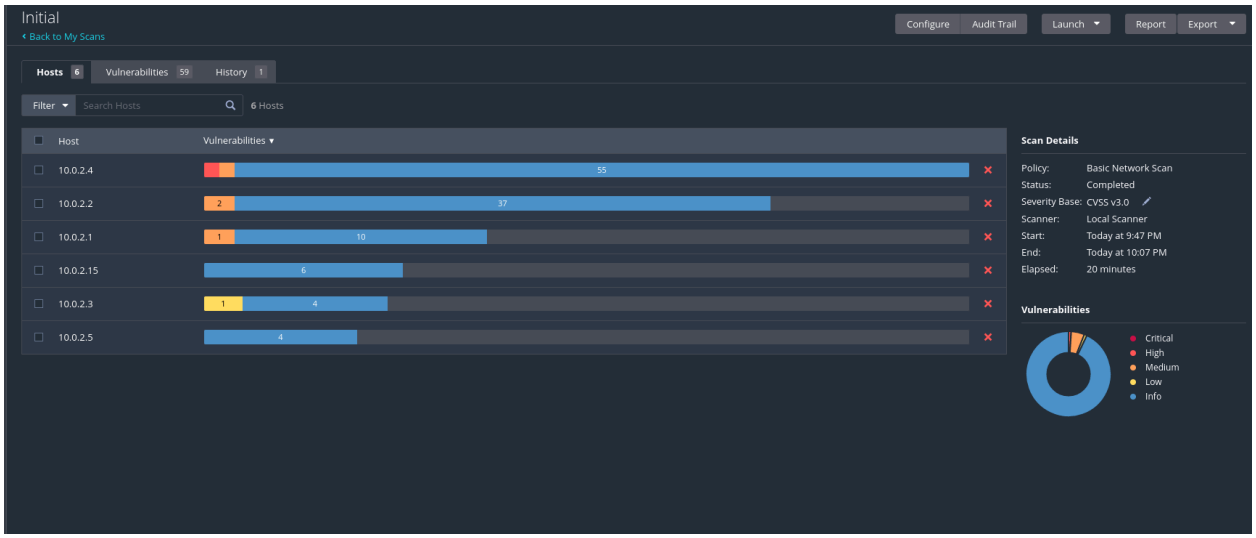

Here, my account is set up and the plugin and feed data has been installed.




In this step, I have created a new scan and filled in the form.




I have launched the scan.




The scan is now complete. I will now explore one of the vulnerabilities.


## Task 3:

```
timothyd@ubuntu:~$ sudo apt install snort -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libdaq2 libdumbnet1 libluajit-5.1-2 libluajit-5.1-common libnetfilter-queue1
  oinkmaster snort-common snort-common-libraries snort-rules-default
Suggested packages:
  snort-doc
The following NEW packages will be installed:
  libdaq2 libdumbnet1 libluajit-5.1-2 libluajit-5.1-common libnetfilter-queue1
```

In this step, I am downloading snort.

```
timothyd@ubuntu:~$ snort --help

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

USAGE: snort [-options] <filter options>
Options:
        -A          Set alert mode: fast, full, console, test or none  (alert file alerts only)
                    "unsock" enables UNIX socket logging (experimental).
        -b          Log packets in tcpdump format (much faster!)
        -B <mask>   Obfuscated IP addresses in alerts and packet dumps using CIDR mask
        -c <rules>  Use Rules File <rules>
        -C          Print out payloads with character data only (no hex)
        -d          Dump the Application Layer
        -D          Run Snort in background (daemon) mode
        -e          Display the second layer header info
        -f          Turn off fflush() calls after binary log writes
        -F <bpf>    Read BPF filters from file <bpf>
        -g <gname>  Run snort gid as <gname> group (or gid) after initialization
        -G <0xid>   Log Identifier (to uniquely id events for multiple snorts)
        -h <hn>     Set home network = <hn>
                    (for use with -l or -B, does NOT change $HOME_NET in IDS mode)
        -H          Make hash tables deterministic.
        -i <if>     Listen on interface <if>
        -I          Add Interface name to alert output
```

Snort has been installed.

```
timothyd@ubuntu:~$ cd ~/Downloads
timothyd@ubuntu:~/Downloads$ ls
2016-04-16-traffic-analysis-exercise.pcap.zip
timothyd@ubuntu:~/Downloads$ unzip ^C
timothyd@ubuntu:~/Downloads$ unzip 2016-04-16-traffic-analysis-exercise.pcap.zip
Archive:  2016-04-16-traffic-analysis-exercise.pcap.zip
[2016-04-16-traffic-analysis-exercise.pcap.zip] 2016-04-16-traffic-analysis-exercise.pcap password:
  inflating: 2016-04-16-traffic-analysis-exercise.pcap
timothyd@ubuntu:~/Downloads$
```

I have downloaded the zip file and unzipped it.

```
timothyd@ubuntu:~/Downloads$ sudo su -
root@ubuntu:~# echo 'alert tcp 91.194.91.203 80 -> $HOME_NET any (msg:"Paypal phishing form"; content:"paypal";
sid:21637; rev:1;)' >> /etc/snort/rules/local.rules
root@ubuntu:~# exit
logout
timothyd@ubuntu:~/Downloads$
```

In this step, I am switching to the root user and echoing the rule into the local.rules file.

```
timothyd@ubuntu:~/Downloads$ sudo snort -c /etc/snort/snort.conf -r 2016-04-16-traffic-analysis-exercise.pcap -q -K none -A console
04/15-15:51:57.730858  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 52.85.82.239:80 -> 172.16.155.149:49252
04/15-15:55:04.445572  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 91.194.91.203:80 -> 172.16.155.149:49269
04/15-15:55:06.015751  [**] [1:1841:5] WEB-CLIENT Javascript URL host spoofing attempt [**] [Classification: Attempted User Privilege Gain] [Priority: 1] {TCP} 91.194.91.203:80 -> 172.16.155.149:49267
04/15-15:55:06.933239  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 91.194.91.203:80 -> 172.16.155.149:49266
04/15-15:59:18.292918  [**] [1:21637:1] Paypal phishing form [**] [Priority: 0] {TCP} 91.194.91.203:80 -> 172.16.155.149:49282
04/15-16:00:48.973352  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 172.217.3.46:80 -> 172.16.155.149:49367
04/15-16:00:49.508881  [**] [1:1852:3] WEB-MISC robots.txt access [**] [Classification: access to a potentially vulnerable web application] [Priority: 2] {TCP} 172.16.155.149:49386 -> 172.217.2.46:80
04/15-16:00:49.749435  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 172.217.2.46:80 -> 172.16.155.149:49386
04/15-16:01:10.826146  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 72.167.2.1:80 -> 172.16.155.149:49395
04/15-16:01:10.888641  [**] [1:2925:3] INFO web bug 0x0 gif attempt [**] [Classification: Misc activity] [Priority: 3] {TCP} 172.217.3.46:80 -> 172.16.155.149:49367
timothyd@ubuntu:~/Downloads$
```

Here we can see the Paypal rule was triggered when running snort against the pcap file.

**Task 4:**

```
timothyd@ubuntu:~$ sudo apt install python3-pip
[sudo] password for timothyd:
Sorry, try again.
[sudo] password for timothyd:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

Here I am installing python.

```
timothyd@ubuntu:~$ pip3 install honeypots
Defaulting to user installation because normal site-packages is not writeable
Collecting honeypots
  Downloading honeypots-0.65-py3-none-any.whl (112 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 112.7/112.7 KB 2.3 MB/s eta 0:00:00
Collecting twisted==21.7.0
  Downloading Twisted-21.7.0-py3-none-any.whl (3.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.1/3.1 MB 20.3 MB/s eta 0:00:00
Collecting pycryptodome==3.19.0
  Downloading pycryptodome-3.19.0-cp35-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (2.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.1/2.1 MB 27.2 MB/s eta 0:00:00
Collecting scapy==2.4.5
  Downloading scapy-2.4.5.tar.gz (1.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.1/1.1 MB 29.7 MB/s eta 0:00:00
```

Here I am installing honeypots.

```
timothyd@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:36:05:5d brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.113/24 brd 10.0.0.255 scope global dynamic noprefixroute enp0s3
       valid_lft 172663sec preferred_lft 172663sec
    inet6 2601:207:182:94a0::dd59/128 scope global dynamic noprefixroute
       valid_lft 223517sec preferred_lft 223517sec
    inet6 2601:207:182:94a0:50f:1b16:310e:cecb/64 scope global temporary dynamic
       valid_lft 300sec preferred_lft 300sec
    inet6 2601:207:182:94a0:637:646c:598:ed54/64 scope global dynamic mngtmpaddr noprefixroute
       valid_lft 300sec preferred_lft 300sec
    inet6 fe80::7784:1d55:1304:596f/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
timothyd@ubuntu:~$ python3 -m honeypots --setup mysql:3306
[INFO] For updates, check https://github.com/qeeqbox/honeypots
[WARNING] Using system or well-known ports requires higher privileges (E.g. sudo -E)
[INFO] Use [Enter] to exit or python3 -m honeypots --kill
[INFO] Parsing honeypot [normal]
{"action": "process", "dest_ip": "0.0.0.0", "dest_port": "3306", "password": "test", "server": "mysql_server", "src_ip": "0.0.
21.709905", "username": "test"}
[INFO] servers mysql running...
[INFO] Everything looks good!
```

Here I am running “ip a” to check my ip address. Then I am setting up a MySQL honeypot.


In this step, I am in my Kali VM making a connection to my Ubuntu VM.


I can see that the attack registered here.