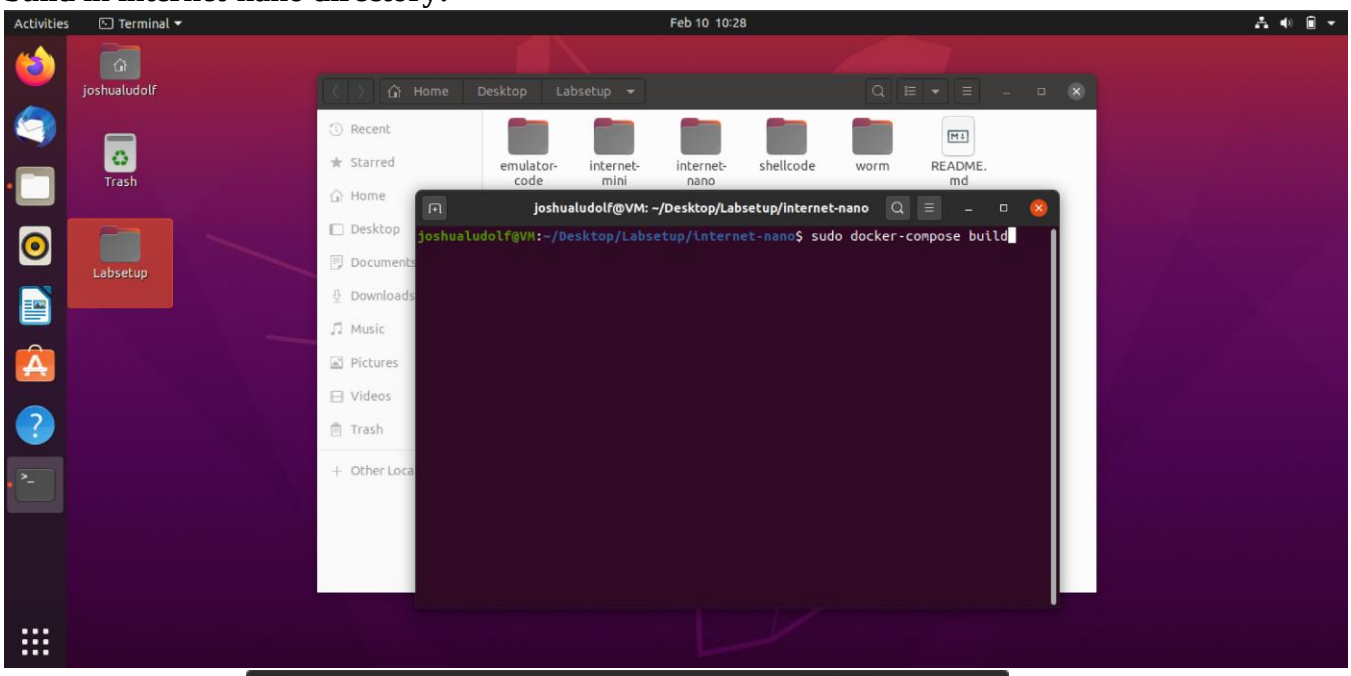
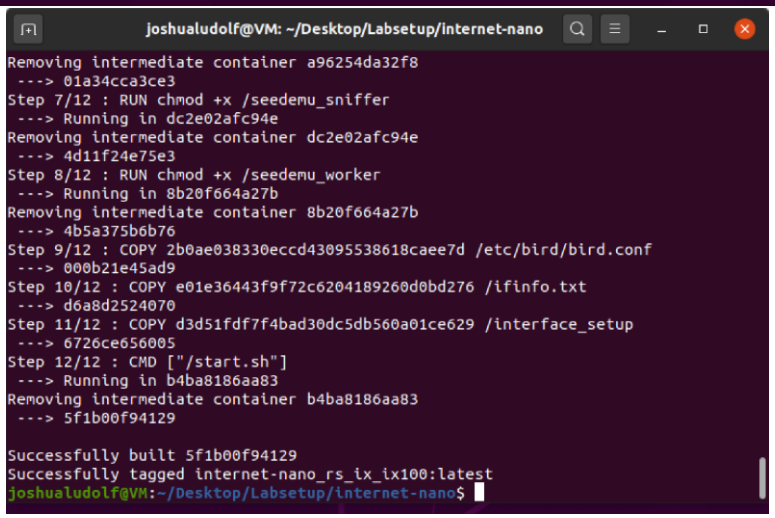
Lab 4: Morris Worm Lab

Joshua Ludolf
CSCI 4321
Computer Security

- **To start this lab, I booted up my vm and typed following command – sudo docker-compose build in internet-nano directory:**



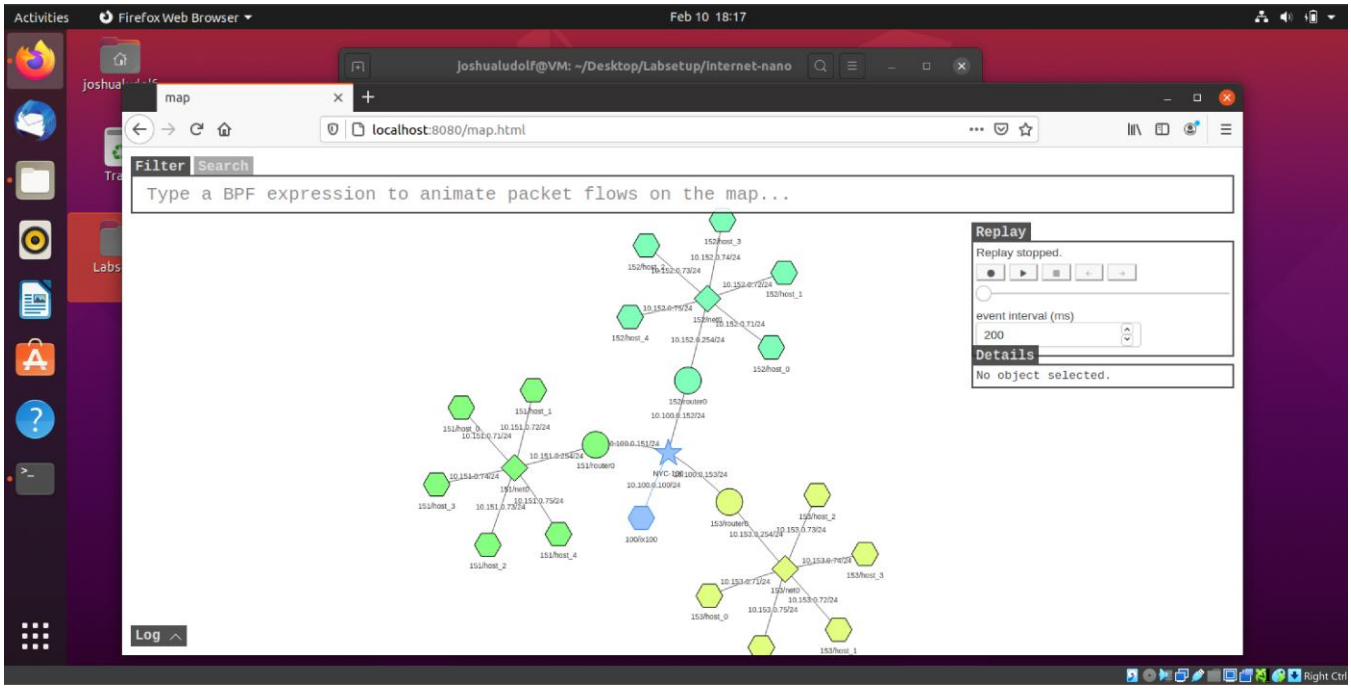


- **From there, I started the docker container using the following command – sudo docker-compose up:**

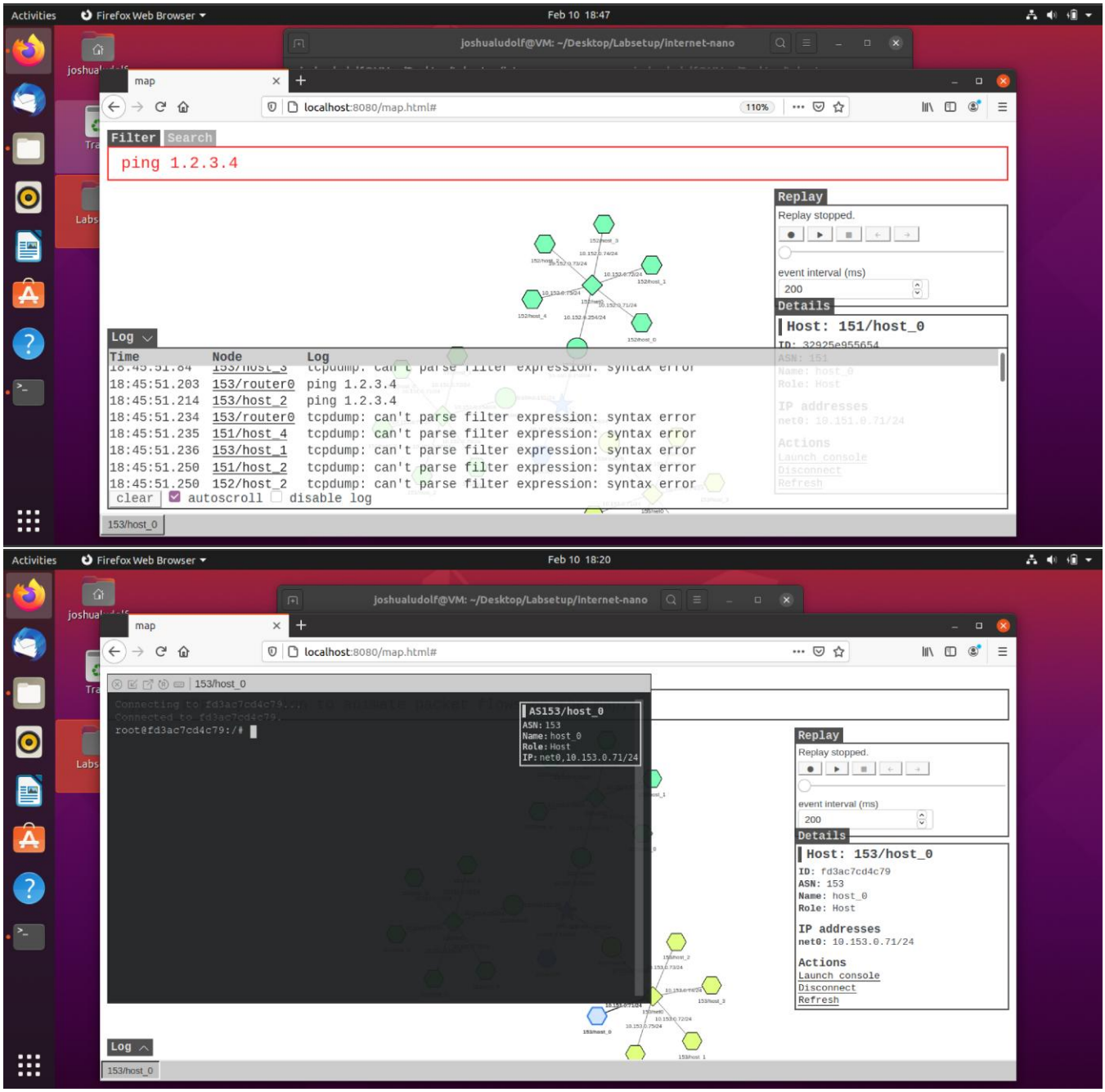
```
joshualudolf@VM:~/Desktop/Labsetup/internet-nano$ sudo docker-compose up
Creating network "internet-nano_default" with the default driver
Creating network "internet-nano_net_151_net0" with the default driver
Creating network "internet-nano_net_ix_ix100" with the default driver
Creating network "internet-nano_net_152_net0" with the default driver
Creating network "internet-nano_net_153_net0" with the default driver
Pulling seedemu-internet-client (handsonsecurity/seedemu-multiarch-map:buildx-la
test)...
buildx-latest: Pulling from handsonsecurity/seedemu-multiarch-map
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
abe80f076312: Pull complete
6908e862c0a6: Pull complete
1de918b5ac72: Pull complete
4f4fb700ef54: Pull complete
```

```
joshualudolf@VM: ~/Desktop/Labsetup/internet-nano
internet-nano_ee6b6326cce7e5be4913cbfc86f3c820_1 exited with code 0
internet-nano_39e016aa9e819f203ebc1809245a5818_1 exited with code 0
internet-nano_morris-worm-base_1 exited with code 0
as153h-host_1-10.153.0.72          | ready! run 'docker exec -it 72ea8a695e70 /
bin/zsh' to attach to this node
as151h-host_3-10.151.0.74          | ready! run 'docker exec -it 6916e8b6e031 /
bin/zsh' to attach to this node
as153h-host_0-10.153.0.71          | ready! run 'docker exec -it fd3ac7cd4c79 /
bin/zsh' to attach to this node
as152h-host_0-10.152.0.71          | ready! run 'docker exec -it c7172040f6d5 /
bin/zsh' to attach to this node
as151h-host_4-10.151.0.75          | ready! run 'docker exec -it 7b785cf424c6 /
bin/zsh' to attach to this node
as151h-host_1-10.151.0.72          | ready! run 'docker exec -it 7903dc1d874e /
bin/zsh' to attach to this node
as153h-host_4-10.153.0.75          | ready! run 'docker exec -it 30517f181868 /
bin/zsh' to attach to this node
as153h-host_3-10.153.0.74          | ready! run 'docker exec -it 1e8eb2db8eac /
bin/zsh' to attach to this node
as152h-host_1-10.152.0.72          | ready! run 'docker exec -it 56e0940b9ddc /
bin/zsh' to attach to this node
as152h-host_2-10.152.0.73          | ready! run 'docker exec -it bccca4e6f39c /
bin/zsh' to attach to this node
```

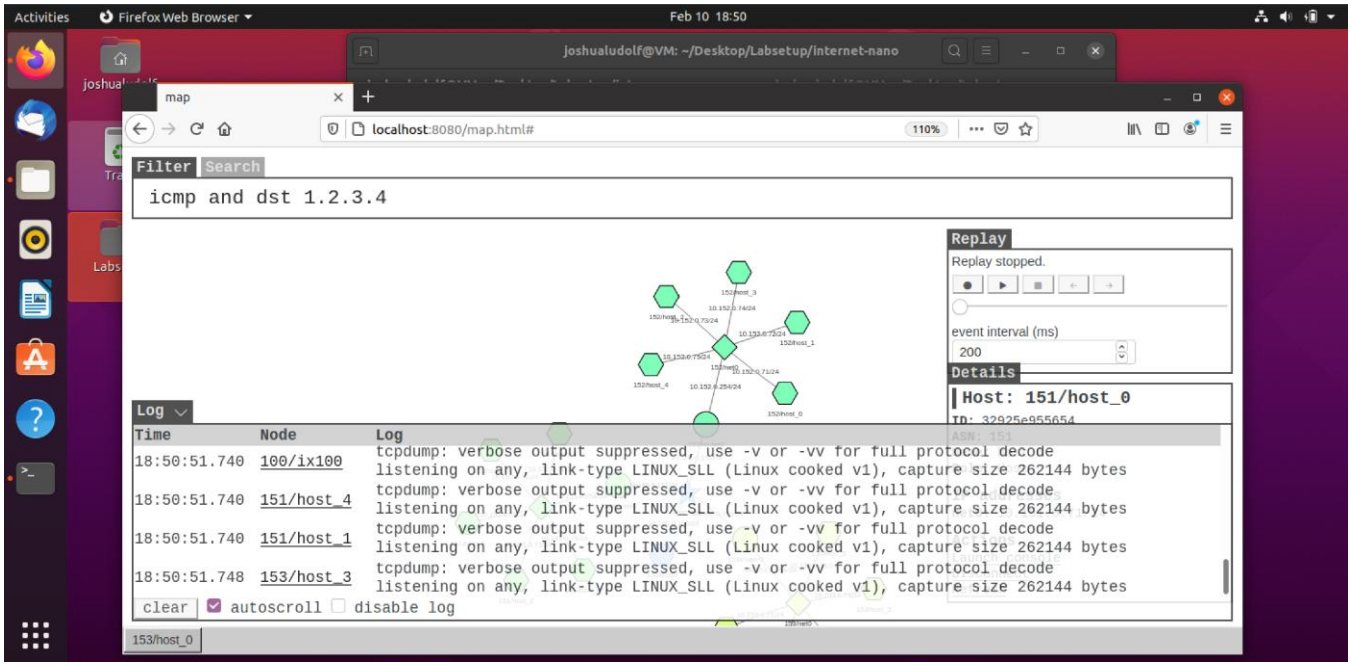
- **Next, I follow the same procedure but in the map directory folder (that is running locally from the docker container so it not local to the linux terminal; that took a sec/guess to understand):**



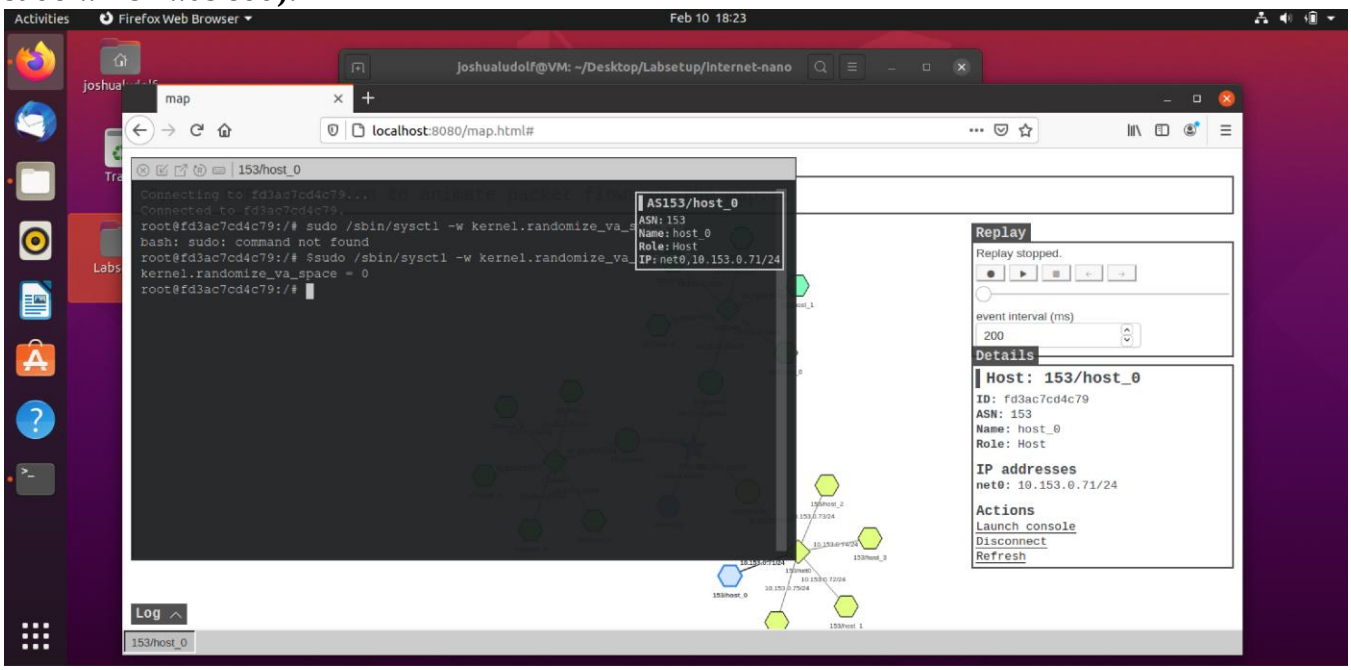
- **I then chose 153/host_0 and opened its terminal, right before that I tried pinging 1.2.3.4 (which got an expected error 😉):**



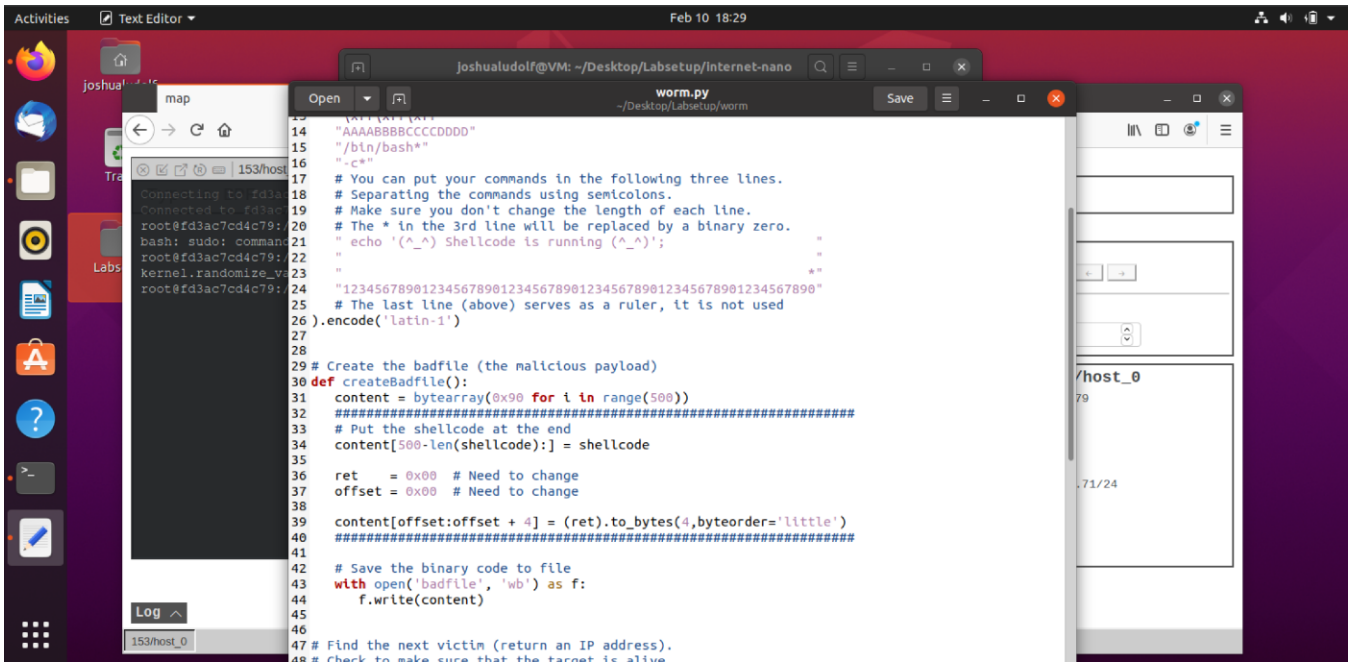
- **Furthermore, I tried the ICMP command:**



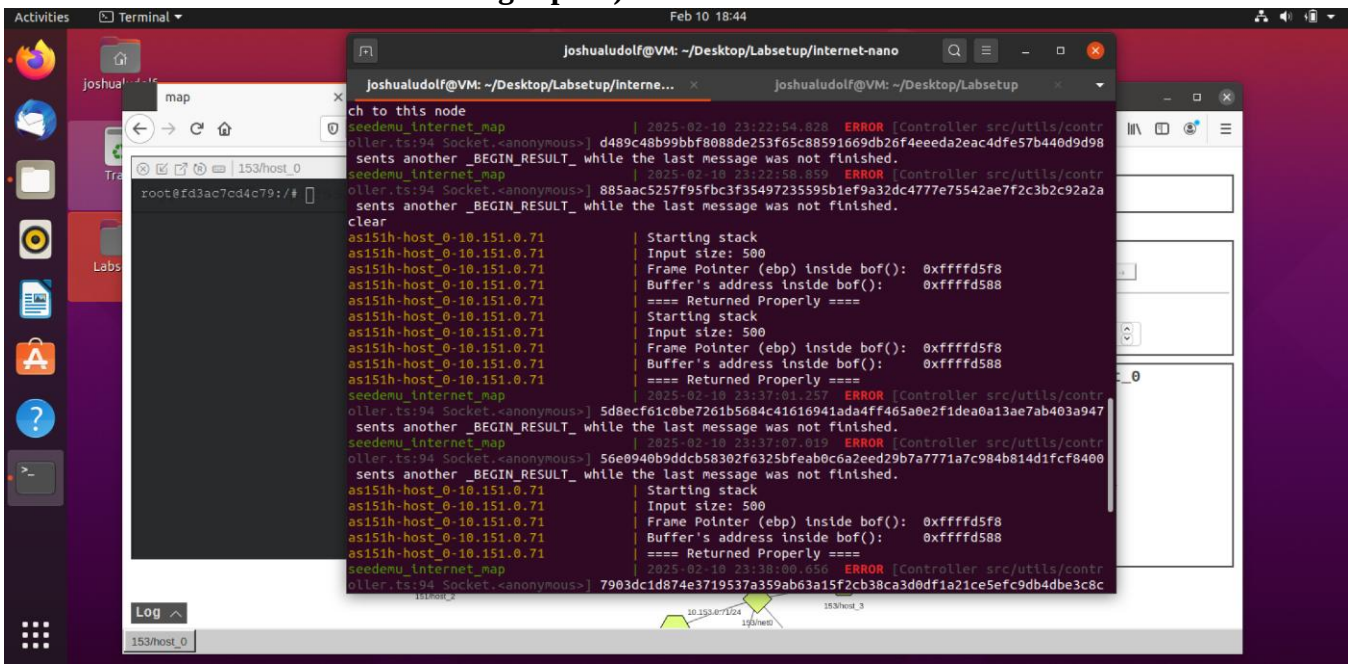
- **At that point I went to original documentation for seed lab and ran following command - sudo /sbin/sysctl -w kernel.randomize_va_space=0 (surprisingly had to put $ in front of sudo which was odd):**



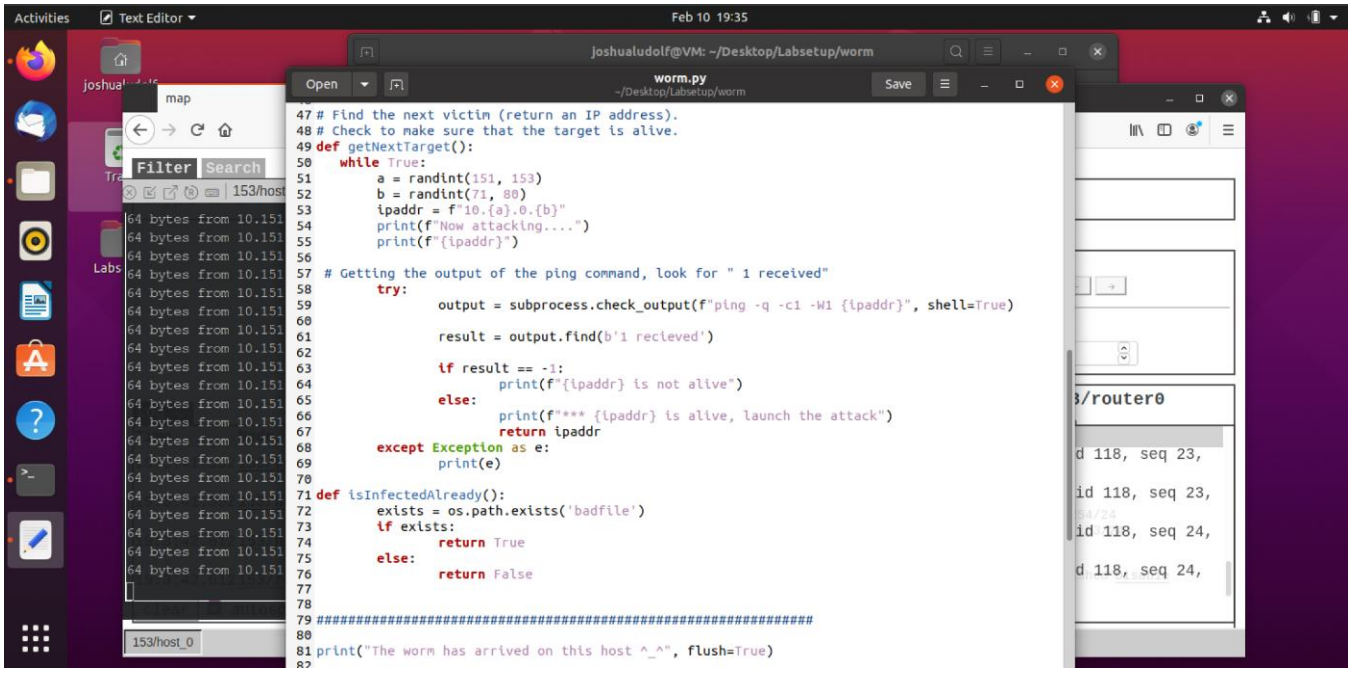
- **From there, I opened the attack code, worm.py and modified it (actually, learned that since I downloaded the file, that all the code was already in it, but I at least examined the code for slightly better understanding [only minor adjustments needed] 😉):**



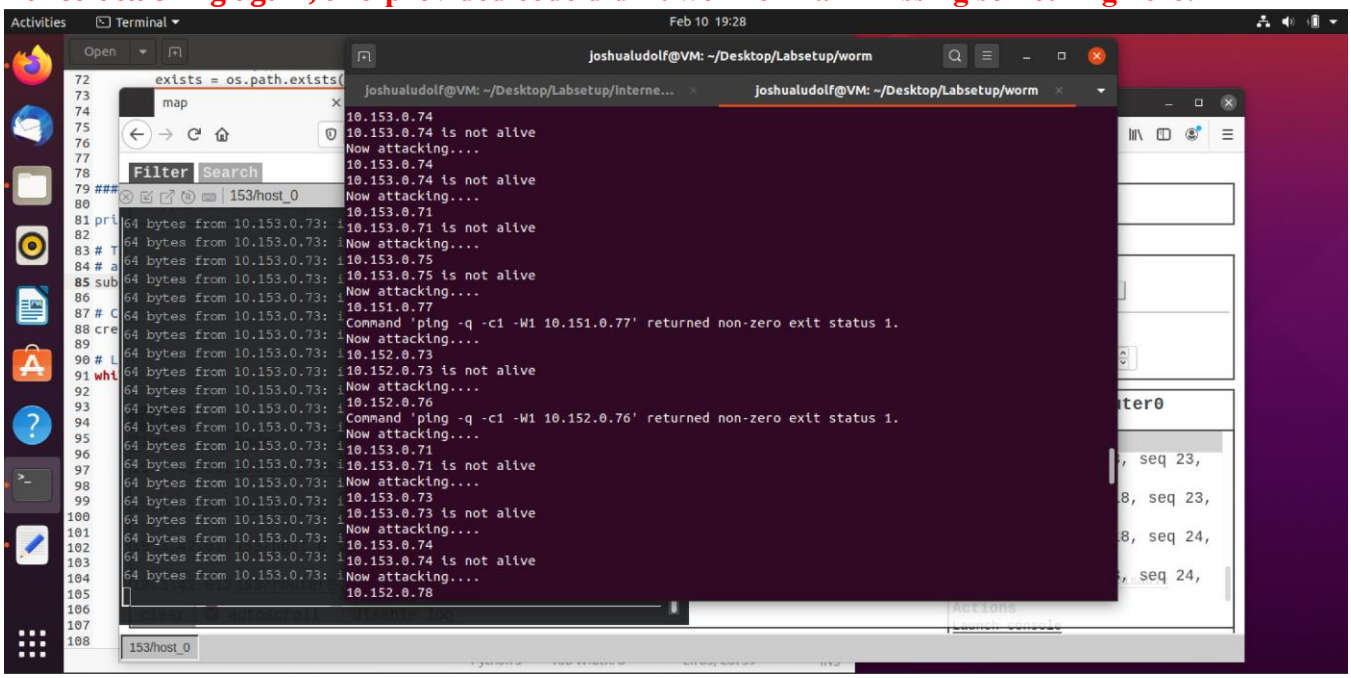
- **When I ran the attack the first time (sadly didn't get smiley face but also didn't know what adjustments to make for the ret and offset, but what I got kinda looked similar to instructions result so kinda on the right path):**



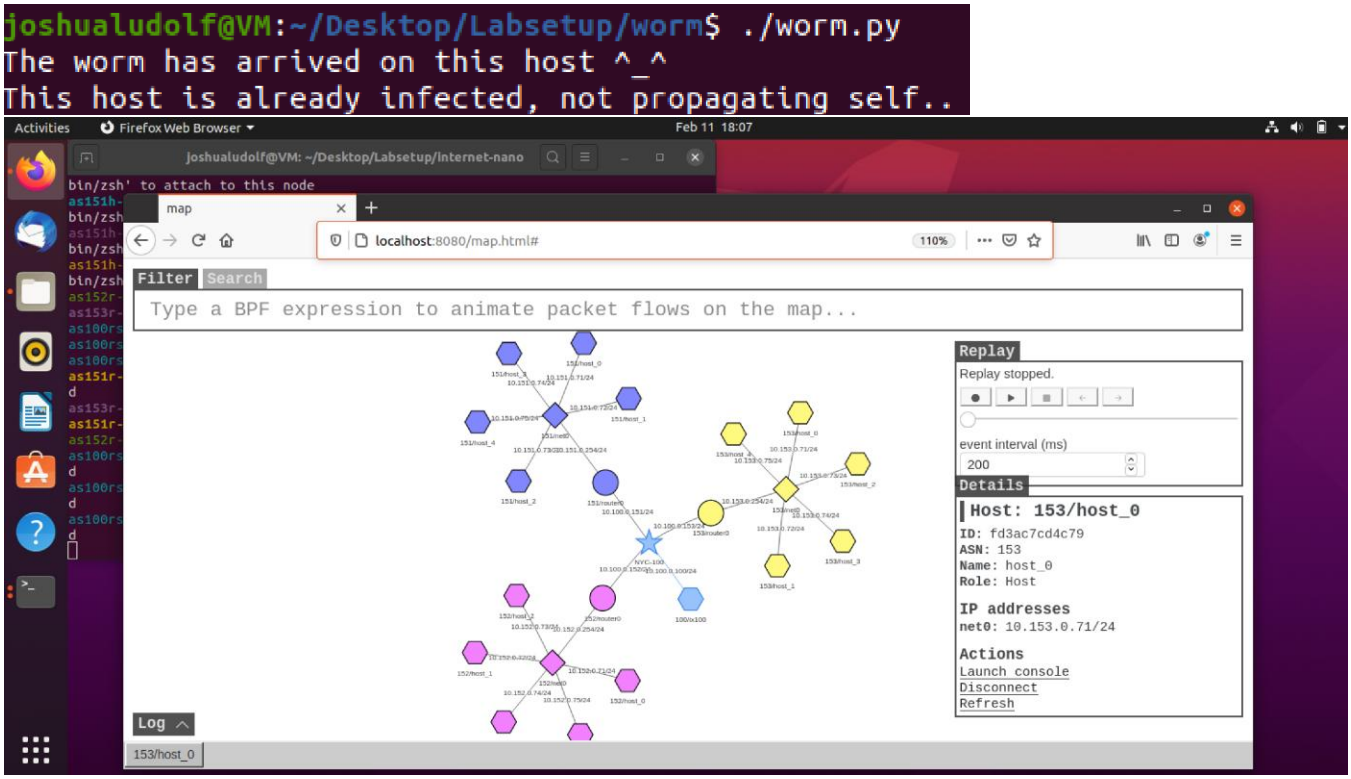
- **Realized that I needed to go back to prior instructions to ping on a host machine (which mine did a blue outline for little bit then stopped for color changing):**



- **I tried attacking again, and provided code didn't work or I am missing something here:**



- **But found someone who performed the lab and edited the worm.py file and apparently the attack was launched successfully but didn't output as it did change colors:**



- **What I learned from this lab:**

In this lab, I learned how to set up and navigate a virtualized environment using Docker. I started by booting up my VM and running docker-compose build and docker-compose up to launch the necessary containers, which helped me understand the differences between executing commands in the local Linux terminal versus inside a Docker container. I experimented with network commands, such as ping and ICMP, and observed the expected failure when attempting to reach a non-existent host like 1.2.3.4. Additionally, I disabled address space layout randomization (ASLR) by running sudo /sbin/sysctl -w kernel.randomize_va_space=0, even though I encountered some unexpected syntax issues along the way. I also delved into the attack code in worm.py, where I examined and made modifications to understand how the payload was constructed and delivered. Despite some initial difficulties with setting the correct values for the return address and offset, the process of debugging and iterating on the code provided me with a deeper insight into practical exploit development and network security challenges. Overall, this lab gave me valuable hands-on experience with both containerized environments and real-world attack scenarios.