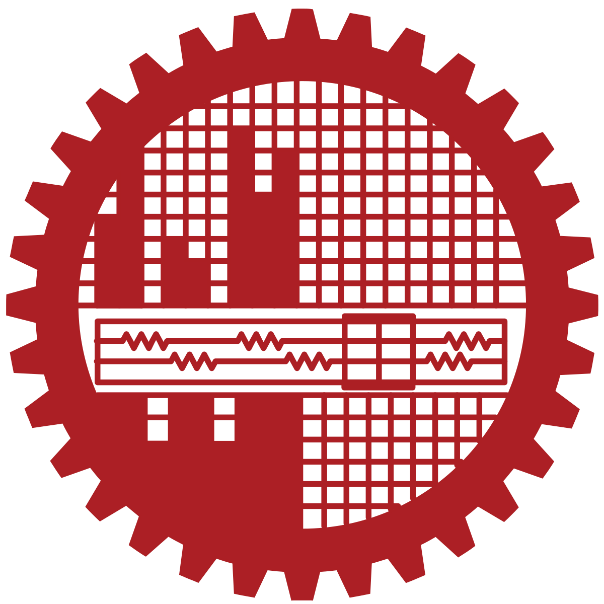
**CSE 406**

**Computer Security Sessional**

**Malware Offline Assignment Report**



Name: Md. Shahrukh Islam

Student ID: 1805098

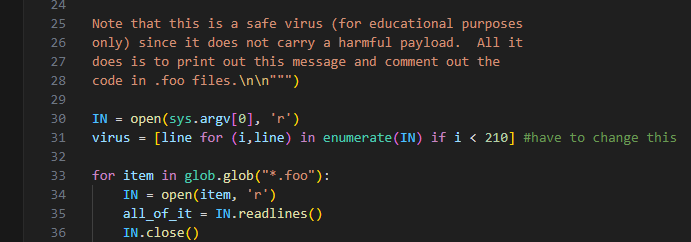
Date: August 4, 2023

**Task 1**

**Turn the FooVirus.py virus into a worm.**

**Solution Approach:**

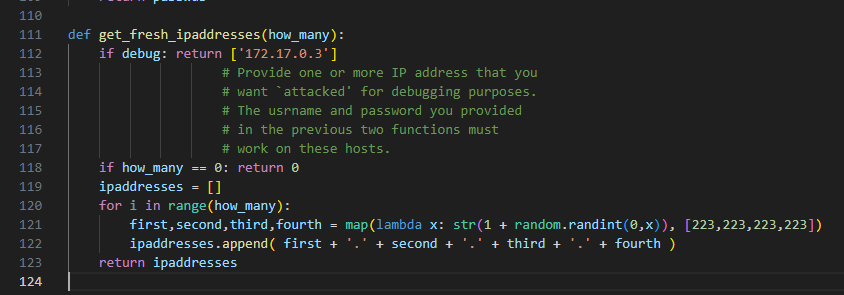
Firstly, the entire code of FooVirus.py was kept. Then networking code for sending the FooVirus into a machine.



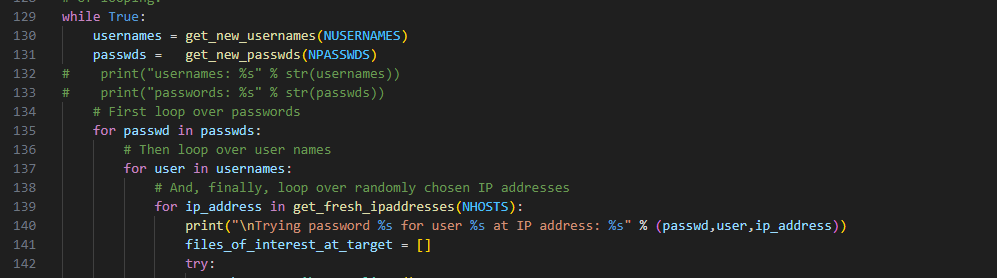
New code is 209 line. So in line 31 it is mentioned. There is nothing change from the FooVirus.py.

For the part, to make this virus as a worm a networking codebase is introduced.

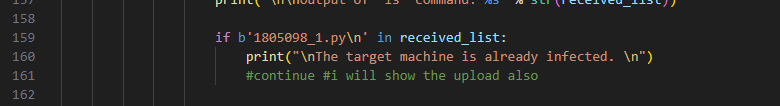
Here a same docker is attacked. So this code is returning same ip address each and every time.



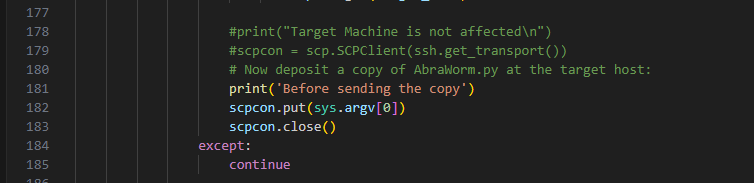
Here loop for ip address is nested. Because many software can detect the attack if this ip is attack much in a very short period of time. So for a good interval ip address loop is inside most.



An intelligent virus doesn’t propagate if that system is already affected. So it is checked that if there is 1805098\_1.py file inside that list or not. An upload mechanism is shown that’s why continue statement is comment out.



Then a copy for this virus is sent to the victim docker.



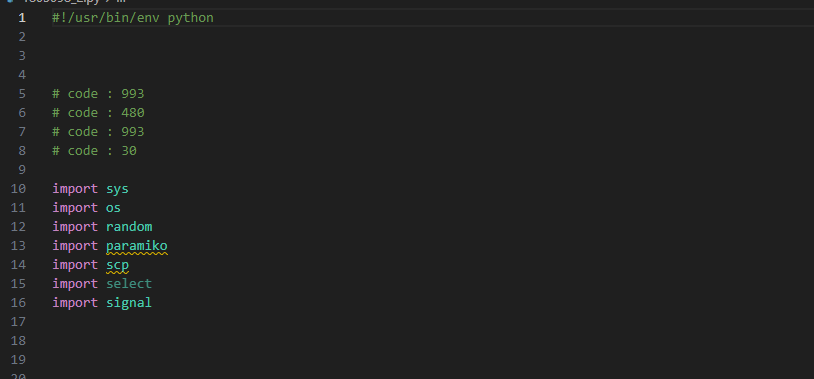
Finally uploaded a copy if we get a sign of our file. This happens when we attack second time and got our sign. Before the first attack all files were safe and that’s why first attack can’t upload it. In second attack in the same system, the upload system will work.

**Task 2**

**Problem:** Modify 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.

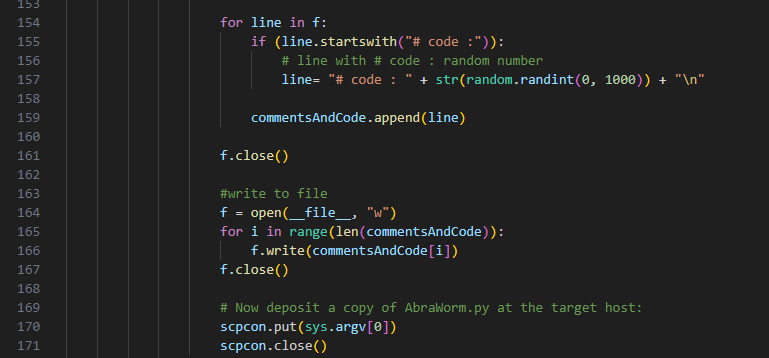
**Solution Appraoch:**

**In the codebase of AbraWorm.py 10 new lines have been introduced. For example:**

****

The line 5-8 is added. And these are comment line. More 6 lines like these are also introduced in different sections of this code.

Then when reading this file if some line starts with “#code :” then after that a new random integer from 0 to 1000 is written. And the current code file is over-written.



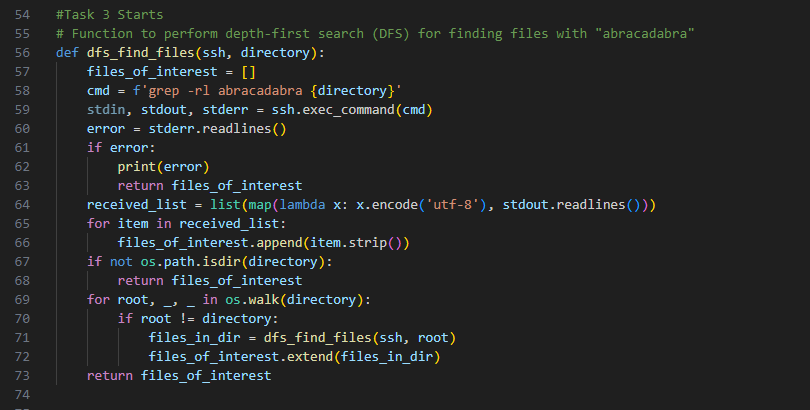
Thus we can get a new file that is different from previous one. Like this 100010 = 1030 combination of this same file is possible.

**Task 3**

**Problem:** Extend the worm code so that it descends down the directory structure and examines the files at every level.

Solution Approach:

A recursive function to look each folder in the directory is introduced:

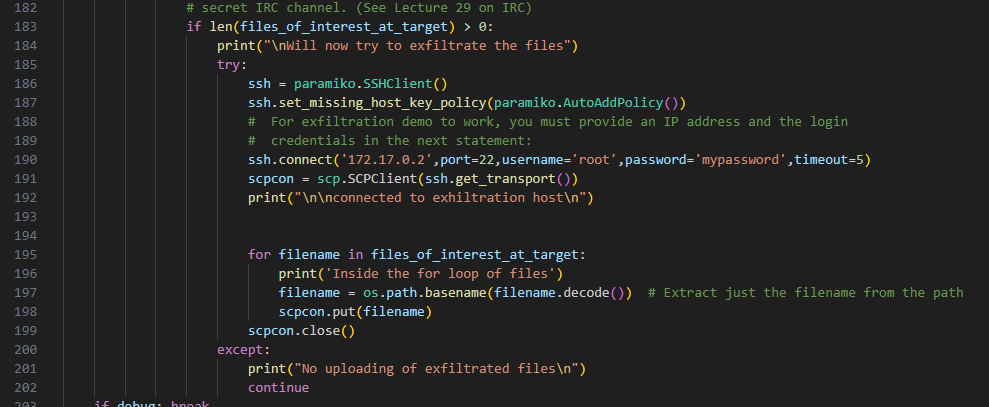


The os.walk() function generates the file names in a directory tree by walking either top-down or bottom-up through the directory tree.

So in root variable we have all subdirectories also including the current directory. Current directory is already expaned. So if root is not directory then a recursive call is introduced. So the loop is intended to traverse all the directories in the target system to find files with the name “abracadabra” using dfs\_find\_files() function.

In files\_of\_interest\_at\_target the whole path of the all target files is kept. But in our local directory only the file has been stored. So we have to filter only the name of the file.

The function os.path.basename() is a built in function of os package. It takes the path and return the only file name.



In the third task the code also changes in each and every run of the file and so sends different code each time. Also it recursively checks the files containing “abracadabra” key word and sends it to a specific ip address.