UNIVERSITY OF WOLLONGONG AUSTRALIA

CSCI369 Ethical Hacking

# Assignment

Due: 11:55 pm 26 October 2025
Total Mark: 100 (30% of Final Mark)

General Instructions: Please read the following instructions carefully.

- You must create a folder for each question. – Create folders named as `Q1, …, Q5`.
- Your Python source codes or text, doc files for each question must be saved in the folder you have created.
- You must install a VirtualBox on your laptop or desktop. In the VirtualBox, you must have at least Kali, Ubuntu and Metasploitable2 virtual machines.
- **You must use Python packages specified in each question.**
- **For Python implementation, using PyCharm is highly recommended.**
- ***Important note***: *You must submit your Python source code with brief readme files (for explaining how to run your program). Not doing so could result in a 5% reduction in the marks for each question.*

1. Writing ARP Spoofer (15 marks)

   In this task, you will write a Python program that allows you to perform an ARP spoofing attack with a single command as follows:

   ```
   sudo python3 arpspoof.py <Victim_IP> <Gateway
   (router)_IP>
   ```

   Your program must use the code snippet we created for Task 1 (Q1) of the Lab 5, which is based on the Scapy package. Note that your program will be checked when the victim is Metaspoitable2 and the attacker is Kali.

2. Writing SSH Brute-forcer (15 marks)

   In this task, you will write a Python program to brute-force passwords to access an SSH server. The user inputs for this program are 1) the target IP address, 2) username and 3) a list of 10 possible passwords. Your program must use the code snippet we created for Task 4 (Q4) of the Lab 6, which is based on the `Paramiko` package. Note that your program will be checked when the target is Metaspoitable2 with `msfadmin` as username.

3. Writing Reverse Shell (30 marks)

   In this task, you will implement reverse shell: Once a victim (a Linux user) runs `revshell.py`, the server gets connection from the victim and will be able to enter Linux commands remotely and run them on the victim's machine. A challenging part of this task is to make the program possible to run the "**cd**" command on the victim machine.

For this task, you will be given `server.py` and `revshell.py`, which you need to modify. (More precisely, you need to complete the functions, `server_run()` and `client_run()` in `server.py` and `revshell.py`, respectively.) `server.py` is a server program that waits for the connection from the victim while `revershell.py` is a piece of reverse shell malware, running on the victim's machine (client). In both programs, the JSON package is employed to transfer a large amount of data seamlessly between the server and the victim (client). Note that **both** of your programs, server.py and revshell.py, will be run on (the same) Kali to assess your programs for the sake of convenience. (That is, you don't need to run `server.py` and `revshell.py` on separate VMs.)

4. Writing Ransomware (20 marks)

In this question, your task is to implement a simple ransomware using Python. (We learned the concept.) The assumption is the following: 1) An attacker breaks into a victim's machine that has OpenSSL installed (For compatibility, use Kali VM.); 2) the attacker put her public key in the victim's machine; 3) the victim has a file named `my_secrets.txt` in his root directory (You can write anything in `my_secrets.txt`.); 4) all the formats for ciphertext outputs should be base64 (human readable).

The ransomware should perform the following:

1) It randomly generates a 16-byte (128-bit) key for symmetric encryption and saves it to a file named key.txt. To generate a key you must use the command: `openssl rand -base64 16`.
2) It also need to generate a public/private key pair (for the attacker).
3) Then it encrypts the file `my_secrets.txt` using the key that the attacker selected in step 1). We call this ciphertext `data_cipher.txt`.
4) The file `key.txt` will be encrypted using the attacker's public key generated in step 2)- We call this `key_cipher.txt.`(Remember the format of the resulting ciphertext should be base64. )
5) The file `key.txt` will be deleted.
6) The file `my_secrets.txt` will be deleted
7) It will finally display a message for ransom payment: "Your file important.txt is encrypted. To decrypt it, you need to pay me $10,000 and send `key_cipher.txt` to me."

Write a Python program that does the above steps. You can use subprocess and/or sys modules to do this task.

5. Gift Voucher Code Cracking (20 marks)

**This is a CTF (Capture-The-Flag) style task. You need to submit a (instead of program codes) report in Word format for this task.

An online shopping retailer runs a server to generate gift voucher codes for customers. More specifically, the server will generate a gift voucher code if it receives a client ID from the customer's machine and sends the generated gift voucher code to the customer. The known technical detail about this system is that the server provides this service using UDP on a port between 12345 and 12500 and uses the MD5 hash function (weak one!) to generate the voucher code.

The gift voucher code has monetary value and is sent to the customer for a certain period only. However, as a hacker, you discovered that the server admin forgot to close the port for the service. You want to generate valid gift voucher codes on your own using many client IDs you collected from information gathering.

Your task is to answer the following questions as the hacker.

a) Run the server program provided with this specification on the Ubuntu VM by running ./`executable_server` on the terminal. (Of course, you need to make the file executable. Note that `executable_server` is for VirtualBox and `executable_server2` for UTM.) Then, use an appropriate tool you learned in CSCI369 to identify the open port between 12345 and 12500 for this service. (Note that the service is based on UDP, so you need to find a right option for that scanning. This option was not covered explicitly, but you can find it easily.) (5 marks)
b) Assume that you use your 7-digit UOW student number as a client ID. Based on the port identified from a), use a netcat command to obtain a gift voucher code for your client ID (i.e. your UOW student ID). (5 marks)
c) It is known that the gift voucher code is generated by the MD5 hash function, taking A||ClientID||B as input, that is,
   *VoucherCode*=MD5(A||ClientID||B)
   where || indicates append, A=[aa,ab,…,az,ba,bb,…, ,zz] is a set of two lowercase alphabet characters; B=[##,^@,…,^&] is a set of two symbols (allowing to have two same symbols such as ##); ClientID is your 7-digit UOW student number, such as 1234567. **Using the Hashcat (**https://hashcat.net/hashcat/**) and Crunch (**https://www.kali.org/tools/crunch/**) tools** (both are available on Kali)**, find the two-alphabet character from A and the two-symbol character from B that the server used to generate a gift voucher code.** (20 marks)

Create a text file called "Q5_answers" and write your answers there. You must explain how you get the answers in detail. Answers without detailed explanation may result in 0 mark (even if they are correct.)

**How to submit**

Put your folders Q1,…,Q5 to one folder named as your UOW student number, e.g. 5284611. 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.