CS2107 Assignment 2

Last Updated: 27 March 2023

Introduction

This assignment takes the form of an information security capture-the-flag (CTF) style competition. In a CTF, participants solve problems involving security weaknesses to bypass defences to obtain a sensitive piece of information called the `"flag"`.

In this assignment, participants are exposed to some of the common skills required to play in these competitions. When using the Assignment Platform, do not change your username. For password reset, it may take up to 5 working days so do use a secure yet memorable password.

Acknowledgements

This assignment is a collective work of present and past teaching assistants, including Devesh Logendran (AY22/23), Sean Tay (AY22/23), Akash (AY22/23), Kel Zin (AY22/23, AY21/22), Weiu Cheng (AY22/23, AY21/22), Wen Junhua (AY22/23, AY20/21), Shawn Chew (AY 21/22), Chan Jian Hao (AY21/22), Ye Guoquan (AY21/22), Debbie Tan (AY20/21), Jaryl Loh (AY20/21, AY21/22), Chenglong (AY19/20), Shi Rong (AY17/18, AY19/20), Glenice Tan (AY19/20, AY18/19), Ngo Wei Lin (AY19/20, AY18/19), Lee Yu Choy (AY20/21, AY19/20, AY18/19, AY17/18), Nikolas Tay (AY 16/17) and Jeremy Heng (AY 16/17).

Grading Scheme and Due Date

This is an individual assignment. You are allowed to post questions on the Canvas Discussions forum but ensure that the questions do not ask for the solution. Additionally, do not post the answers to the challenges.

This assignment is worth 10% of the grade for the entire module. Assignment 1 is divided into the following sections:

1. **Easy (60 points):** Answer all challenges.
2. **Medium (60 points):** Answer all challenges.
3. **Hard (30 points):** Answer all challenges.
4. **Bonus (30 bonus points):** Answer all the challenges to get bonus marks. Each bonus question can replace 1 other question.

The maximum number of points that can be obtained in this assignment is **150**. Solving the other bonus challenges can help you earn additional bonus points. Note that any bonus points earned in this assignment can be used, if needed, to top up your the 2 CTF assignments (20%). **There are no partial marks.**

The assignment is due **14 April 2023 (Friday), 2359 HRS**.

Penalties

Late submission of challenges

Score penalties will apply for late submissions:

- Late up to 12 hours beyond due date: **10% penalty** to total score obtained
- Later than 12 hours but up to 36 hours beyond due date: **20% penalty** to total score obtained
- Later than 36 hours but up to 72 hours beyond due date: **30% penalty** to total score obtained
- 72 hours beyond the due date: **Submissions will not be entertained after 17 April 2023 (Monday), 2359 HRS**

Other Penalties

Full marks for this assignment is **150**.

1. Submission of past flags: -10 pts
2. Late submission of Writeup: -10 pts
3. No source code: -10 pts
4. Unclear Writeup:
    - Interview to explain solve
    - Unable to explain = -30%
5. Blank Writeups: -40%
    - Will also be asked for interview
    - Unclear writeup deduction will also apply

Note that submitting a late flag beyond the due date will make your whole submission be considered as a late submission, and the mentioned score penalty scheme applies to your total score obtained.

Contact

Please direct any inquiries about the assignment to

1. devesh.logendran@u.nus.edu (Devesh Logendran)
2. sean.tay@u.nus.edu (Sean Tay)
3. wen_junhua@u.nus.edu (Wen Junhua)
4. c.akash@u.nus.edu (Akash Chandrasekaran)

Note that the TAs will **not** be debugging your code, but will only be around to discuss high level ideas. Do allow 3 working days for replies. Discussion on forums are highly encouraged.

Rules and Guidelines

PLEASE READ THE FOLLOWING BEFORE BEGINNING

1. You are required to log in to CTFd (accessible only within NUS SoC Network) to submit flags.

2. You are **required** to upload the required files **separately** to the "Assignment 2" folder on Canvas before the given deadline.

    - A **write up** documenting the approach you took in solving every problem. This must be in PDF format with the following filename format: **StudentID_Name_WU.pdf** (e.g. A01234567_Alice Tan_WU.pdf)

    - All source codes and scripts that you used while solving the problem, if any. Submit each script as a **separate file** named after the challenge it applies to.

    - **Note** that grades are not directly determined by this writeup. However, your writeup should **sufficiently share the approach** that you took in solving every problem. Screenshots may be helpful in showing your steps too. If there are suspicion on plagiarism, your writeup may be analysed and you may need to be interviewed by the teaching team to explain your steps. This writeup also serve as proof of your work in case submission server malfunctions.

3. Do not attack any infrastructure not **explicitly authorised** in this document.
4. Multiple flag submission is permitted on the scoring platform without any penalty, but **no bruteforcing of flag submission on the server** will be tolerated.
5. Work **individually**. Discussion of concepts on the forum is allowed but refrain from posting solutions. The university takes plagiarism very seriously. Any sharing of answers detected will be reported and disciplinary actions will be taken.
6. Students may be randomly selected to satisfactorily explain how they obtain their flags;or else a zero mark will be given on their unexplainable challenges.
7. The skills taught in this assignment are not to be used on any system you do not own or have express permission to test. This is a **criminal offence** under the Singapore Computer Misuse and Cybersecurity Act.
8. All challenges have a solution. They are guaranteed to be solvable with assistance of the internet and some research.
9. Ask the TAs for assistance only after you have exhausted every other avenue of self-help.
10. Every challenge will contain a flag and will provide the accepted flag format. Please ensure your submissions meet the flag format stated **exactly**. This means include the `CS2107{}` portion unless otherwise stated.
11. The challenges are tested from the NUS WiFi within the School of Computing and outside of NUS. Connectivity cannot be guaranteed anywhere else. **SoC VPN is required** if you are outside of school network.

One of the most important skills in the information security field is the skill of seeking an answer independently. It is expected that the participant be able to utilise resources discovered through Google or any other search engine to achieve the tasks.

While the challenges might not be covered in entirety in class, the topics in the assignment are very applicable to security problems in real life. In the long run, the practical skills gained would benefit participants immensely.

Academic Honesty

NUS students are expected to maintain and uphold the highest standards of integrity and honesty at all times. As this is an **individual assignment**, please refrain from any forms of academic dishonesty.

If any form of plagiarism or cheating is found, you will be penalized and be subject to disciplinary action by the University. You may read more about NUS Student Code of Conduct here.

Linux Environment

A Linux system is crucial for solving some of the challenges, the challenges in this section will prepare you for the more advanced sections by presenting some elementary tasks to solve. It is expected that the participant has rudimentary proficiency in using a Linux system that can be gleaned by reading the tutorial at this link: https://www.digitalocean.com/community/tutorials/an-introduction-to-the-linux-terminal.

However, more knowledge might be needed, and it is expected that the participant do some self-exploration.

Do note that you should use a 32-bit / 64-bit Linux environment to aid you in completing some of the challenges. Please also take note that if you are running 64-bit Linux, you may need to run the following commands in Linux to run 32-bit binary executables:

```
sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install -y libc6:i386
```

The nc Command

Throughout the assignments, if you see challenge with `nc aaa.bbb.ccc.ddd xxxx`, then it means that the challenge is hosted on the `aaa.bbb.ccc.ddd` server on `xxxx` port.

You can connect to the server by using the `nc` command in your terminal. In short, you can just copy & paste `nc aaa.bbb.ccc.ddd xxxx` and run it directly.

If you wish to host a TCP server locally, you can use ncat `ncat -lvk -p 15000 -e "python3 main.py"`

Then connect to it with `nc localhost 15000`

Python3 Cheatsheet

Some challenges in the assignment might require some scripting to solve. Although you can use any programming languages you prefer, we recommend Python3.

To dynamically with interact with TCP server, you can use pwntools

```
from pwn import * # Import pwntools

r = remote("123.123.123.123", 15000) # Connect to 123.123.123.123 at port 15000

s = b'abcde'
r.sendline(s) # Send bytes s to the server
r.sendafter(b'message:', s) # Send bytes s after received bytes 'message:'

r.recvline() # Receive a line from the server
r.recvuntil(b'Nonce: ') # Receive until the bytes 'Nonce: ' from the server
r.recvall() # Receive all bytes until EOF

r.interative() # Change to interative mode
```

Note that all the received message are in bytes. So you might to some conversion if necessary.

You can also change to debug mode with

```
r = remote("123.123.123.123", 15000, level='debug')
```

Here's a link to a cheatsheet: https://gist.github.com/DavidTan0527/43edbf49fc550100a5a88d23627480ff

GDB Cheatsheet

To aid in solving System Security (also known as binary exploitation) challenges, we've also prepared a GDB (GNU DeBugger) cheatsheet here: https://gist.github.com/Enigmatrix/89b09b4c97d541df3dd9e0d8ace9ed1a

The purpose of GDB is to monitor the state of the program (directly view memory and register values, pause at certain conditions etc) so that you can see whether the input you have send is correctly influencing the execution of the program.

Easy Challenges (60 marks)

Answer **all** challenges.

E.1 Sanity Check (15 mark)

A flag, written in our flag format, is placed somewhere in the assignment instruction file.

Try to find and submit it!

Flag format: `CS2107{...}`

Author: Akash

E.2 Baby XSS (15 marks)

An introduction to Cross Site Scripting

Website

Author: Junhua

E.3 The Inspector (15 marks)

I have a feeling something is hidden in this website. Can you find it?

Website

Author: Junhua

E.4 Wiresharked (15 marks)

I've intercepted communication between the infamous Greyson Caterine and Mallory. I wonder if there's anything interesting amidst all the noise!

You might want to install wireshark to get started.

Author: Sean

Medium Challenges (60 marks)

Answer **all** challenges.

M.1 JsonCookies (15 marks)

The owner of the website is known to use weak passwords for his JWT Tokens. Can you find the password and get access to the flag?

Website

Author: Junhua

M.2 Over The Moon (15 marks)

If you are high enough (in altitude), you can get my flag! (no it's not the US flag).

NOTE: This challenge is not released yet! We are using another platform to host this challenge. We will announce the release once it is up.

Author: Akash

M.3 UDP Viewer (15 marks)

I made a nice interactive viewer for UDP Packets that calculates the checksum, but is there some bug inside?

P.S. There are (free) hints available on the CTFd

Author: Akash

M.4 Kid SQLi (15 marks)

I really want to check out the secret behind this login page. I used to be able to do it because the admin wrote code that was vulnerable to SQL injection like this:

```
c.execute("SELECT username, password, role FROM users WHERE username = '" +
username + "' AND password = '" + password + "'")
```

But it seems like the admin added some checks to defeat my payload. Maybe they can be bypassed?

Website

Author: Devesh

Hard Challenges (30 marks)

Answer **all** challenges.

H.1 Crypto Bro Bank (15 marks)

Hi, I made a super-AI bank teller for my totally legit bank. I dare you to steal my money and leak my secrets!

P.S. There are (free) hints available on the CTFd

Author: Akash

H.2 SVB (15 marks)

Those CHEATS at Shady Valley Bank have fleeced me out of my money!!! Luckily I have a mole who snuck me a copy of the source code to their website. See if you can find any vulnerabilities that can let us reclaim some of their ill-gotten wealth. We have some social engineering experts on our side so we can make the admin open a page with some exploit code if need be...

Website

NOTE: To simulate the process of client-side exploitation, we have set up a bot where the website admin (who is logged into the website) will automatically visit any URL provided. Submit a URL for the admin to visit like so: http://cs2107-ctfd-i.comp.nus.edu:5004/visit?url=[URL to visit]

The admin will access the web page through the URL http://svb:5003 This will be important for your payload.

Author: Devesh

Bonus Challenges (Bonus Marks)

Answer **all** challenges to get bonus marks.

B.1 manual

What if I give you an obvious vulnerability, but there is no way to exploit it? Come, give it a try!

Author: Akash

B.2 Part-Time Injector

Can you use SQL injection to get hidden data from the database if none of the query results get refelcted onto the page? Try getting the flag from the table `flag` under the column `flag`.

Server: Website

Author: Devesh

CS2107{let_the_games_begin_part_2}