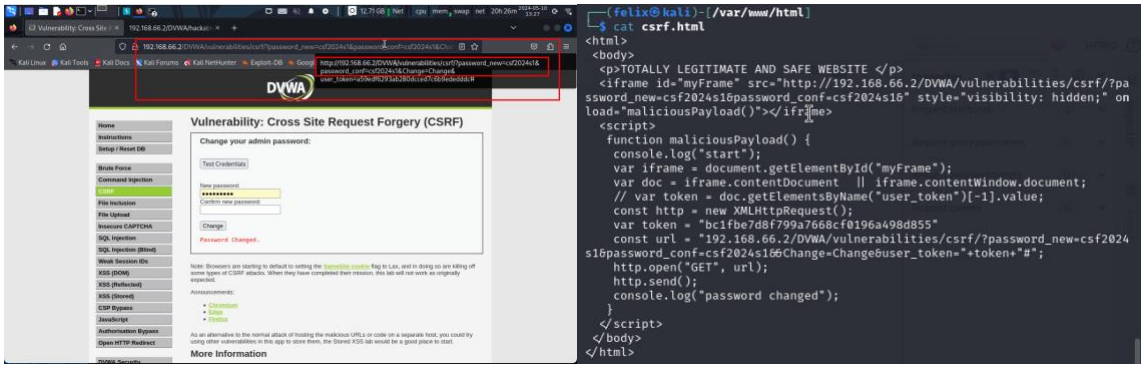
Assignment 0x05 - Advanced Web Exploits and Digital Forensics

Part 1 - Advanced Web Exploits

q1. [1 point] - When on the high-security setting of DVWA, a unique ANTI-CSRF token is created each time the password change page is accessed, as shown in the workshop. To launch a CSRF attack in this case, we first need to steal the token. Create an HTML (name it: 'csrf.html') file that can steal the token from the DVWA CSRF page http://[hacklabvm_ip]/DVWA/vulnerabilities/csrf and change the password to 'csf2024s1'. You can use the template here. Show the content of your csrf.html file.

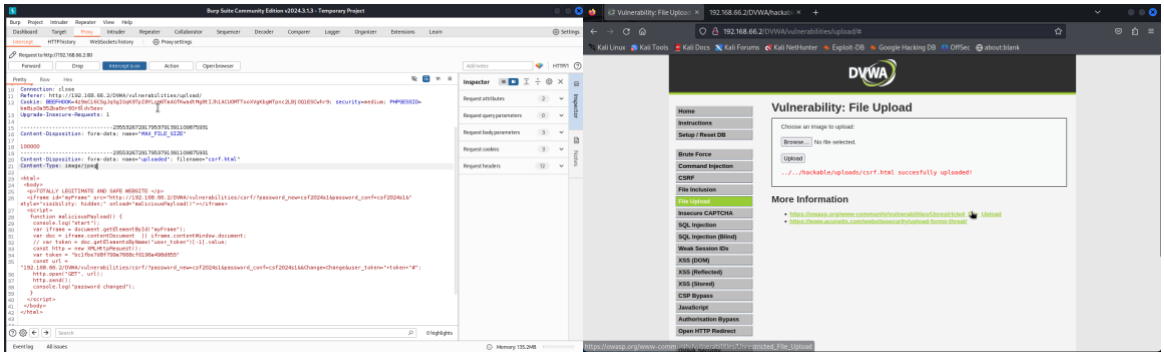
[HINT] To properly craft the malicious HTML file, you will need to take note of the request sent by DVWA when the change password is triggered.



Copy paste part of the URL while changing the password, set the “password_new” and “password_conf” parameters to “csf2024s1” which is in the source field of “iframe”.

q2. [1 point] - Set the DVWA security level to MEDIUM. Upload the csrf.html file to the "hackable/uploads/" folder. Provide details of the steps you use to upload the HTML file with the security level set at medium.

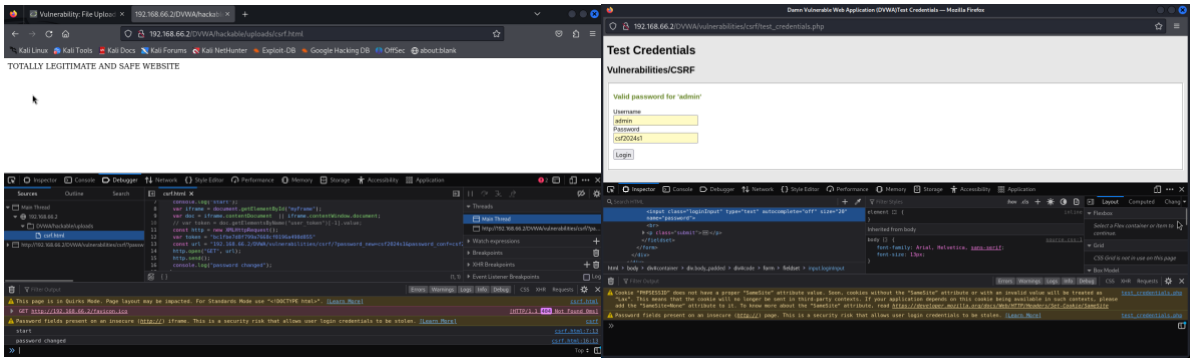
[HINT] Set the DVWA security level to MEDIUM first. At this level of security, you won't be able to upload the HTML file directly as only certain file types are allowed. You may need a proxy intervention.



Totally same as what we did in the workshop, just using the Burp Proxy to capture the upload request and change the content type to "**image/jpeg**" and forward it, it shows success in the file upload page.

q3. [1 point] - Show that after a user visit http://[hacklabvm_ip]hackable/uploads/csrf.html. the password changes to 'csf2024s1'. Explain what happened.

[HINT] To trigger the malicious HTML, take note of the location of where you uploaded your file.



First use the command "console.log(document.getElementsByName("user_token")[0].value)" to get the user token from the console, store it in the csrf.html file.

Finally, open the csrf.html page in the same domain to perform the attack, checking the status by open the console and test the credential in the "Test Credentials" page.

Part 2 - Digital Forensics

Marking rubric for questions 1-3 below: 0.5 point for the final flag, 0.5 point for screenshots of the proof of the process, and 1 point for an explanation of any commands/tools used and the thought process to find the answer.

q1. [2 points] Reversing (1)

- Download this binary . You can run it as "./q1" in Linux
- You ARE NOT allowed to patch this program
- Use Ghidra, Cutter, or Radare2 (or something else) to decompile and deduce the password required for revealing the secret.

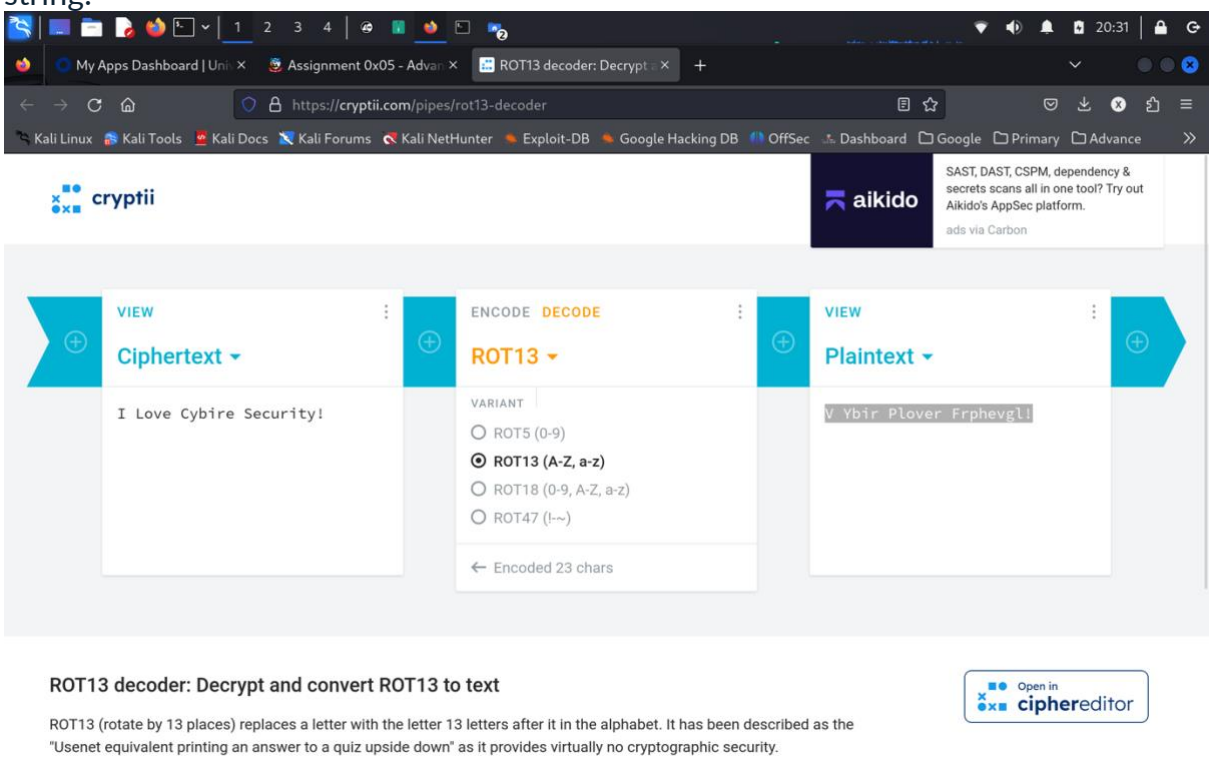
- Get the program to print the secret.

First of all, use cutter to decompile the code, saw that it uses “rot()” function to decrypt the input and compare it to the string “I Love Cyber Security!”.

```
/* jsdec pseudo code output */
/* /Users/felix/Documents/Adelaide_Uni/2024/S1/CSF3308/Assignment/Assi @ 0x98d */
#include <stdint.h>

int32_t main (void) {
    *(var_41ch) = edi;
    *(var_428h) = rsi;
    rax = *(fs:0x28);
    *(canary) = rax;
    eax = 0;
    puts ("What is the password?");
    rax = s1;
    fgets (rax, data.00000400, *(stdin));
    rax = s1;
    rsi = data_00000cd5;
    rdi = rax;
    strtok ();
    rax = s1;
    esi = 0xd;
    rdi = rax;
    rot ();
    rax = s1;
    eax = strcmp (rax, "I Love Cyber Security!");
    if (eax == 0) {
        eax = 0;
        print_secret ();
    } else {
        puts ("Sorry, wrong password...");
    }
    eax = 0;
    rcx = *(canary);
    rcx ^= *(fs:0x28);
    if (eax != 0) {
        stack_chk_fail ();
    }
    return rax;
}
```

So I use the online decrypt website to deal with the string then got the following string:

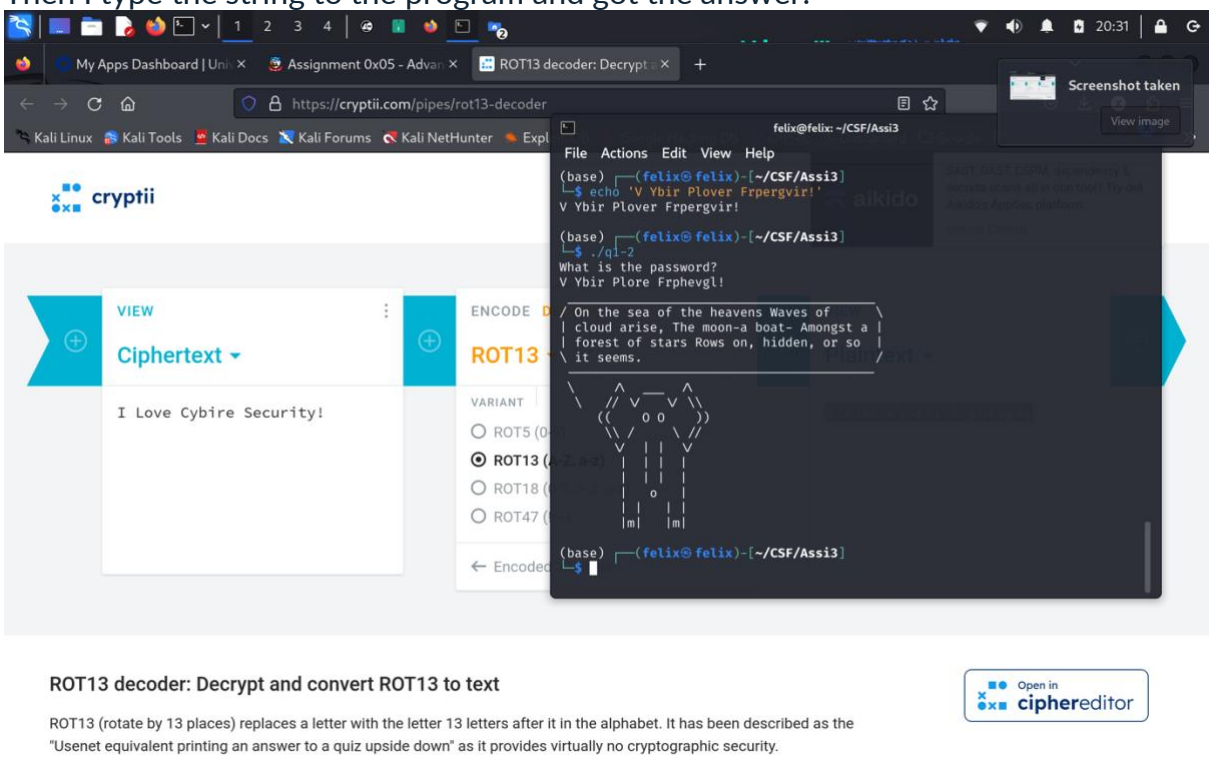


ROT13 decoder: Decrypt and convert ROT13 to text

ROT13 (rotate by 13 places) replaces a letter with the letter 13 letters after it in the alphabet. It has been described as the "Usenet equivalent printing an answer to a quiz upside down" as it provides virtually no cryptographic security.

Open in cipheredtor

Then I type the string to the program and got the answer:



ROT13 decoder: Decrypt and convert ROT13 to text

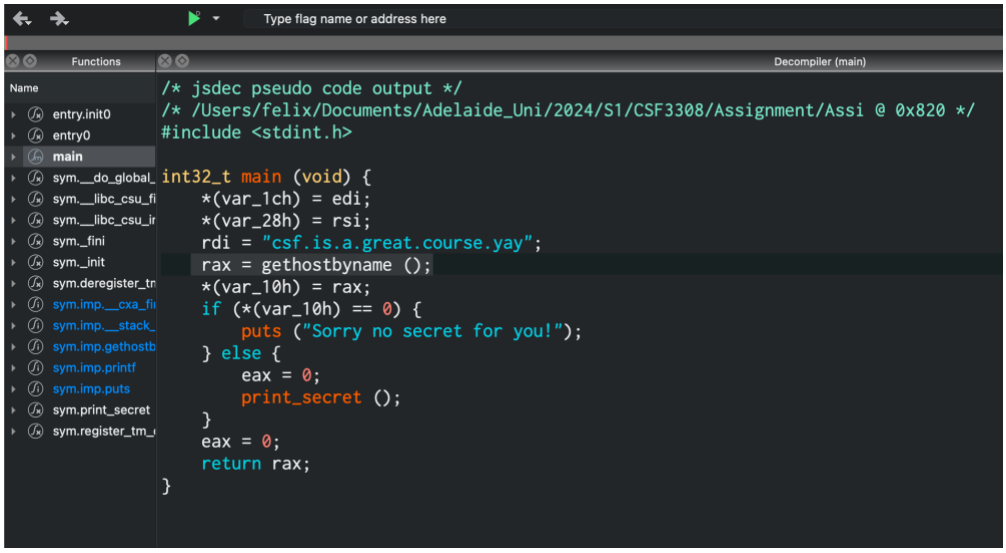
ROT13 (rotate by 13 places) replaces a letter with the letter 13 letters after it in the alphabet. It has been described as the "Usenet equivalent printing an answer to a quiz upside down" as it provides virtually no cryptographic security.

Open in cipheredtor

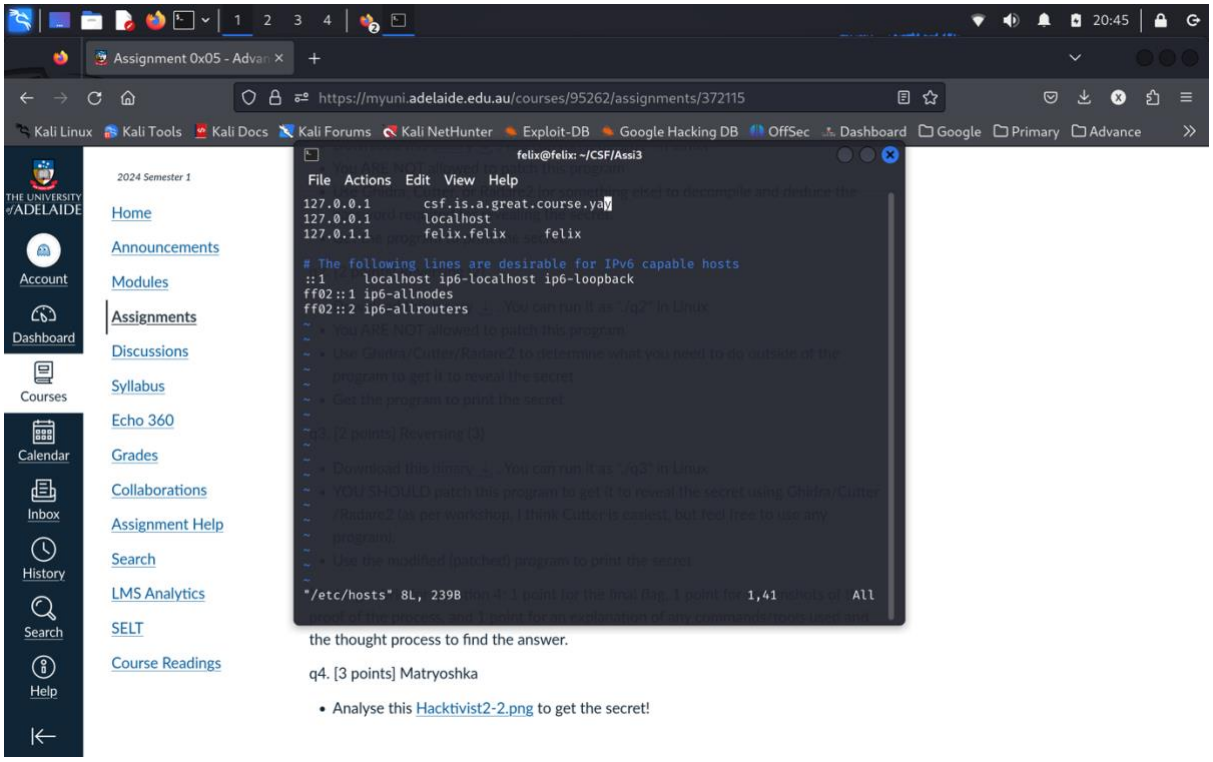
q2. [2 points] Reversing (2)

- Download this binary Download binary. You can run it as "./q2" in Linux
- You ARE NOT allowed to patch this program
- Use Ghidra/Cutter/Radare2 ro determine what you need to do outside of the program to get it to reveal the secret
- Get the program to print the secret

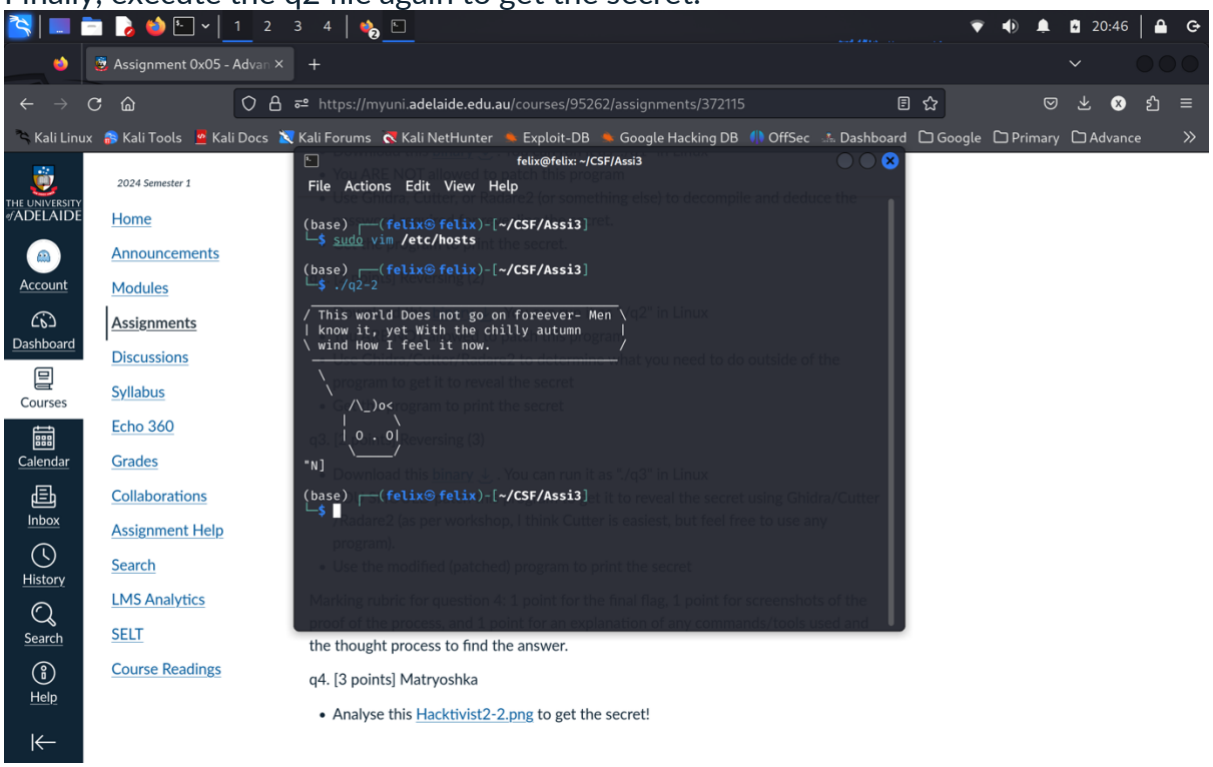
Firstly, use cutter to decompile the code as usual, saw that there is a function called "gethostbyname()", then find information about how to modify the host name of the computer.



Then finding that the way to modify it is to modify the “/etc/hosts” file, adding the line shows in the decompiled file “csf.is.a.great.course.yay” to the first line of the host file:



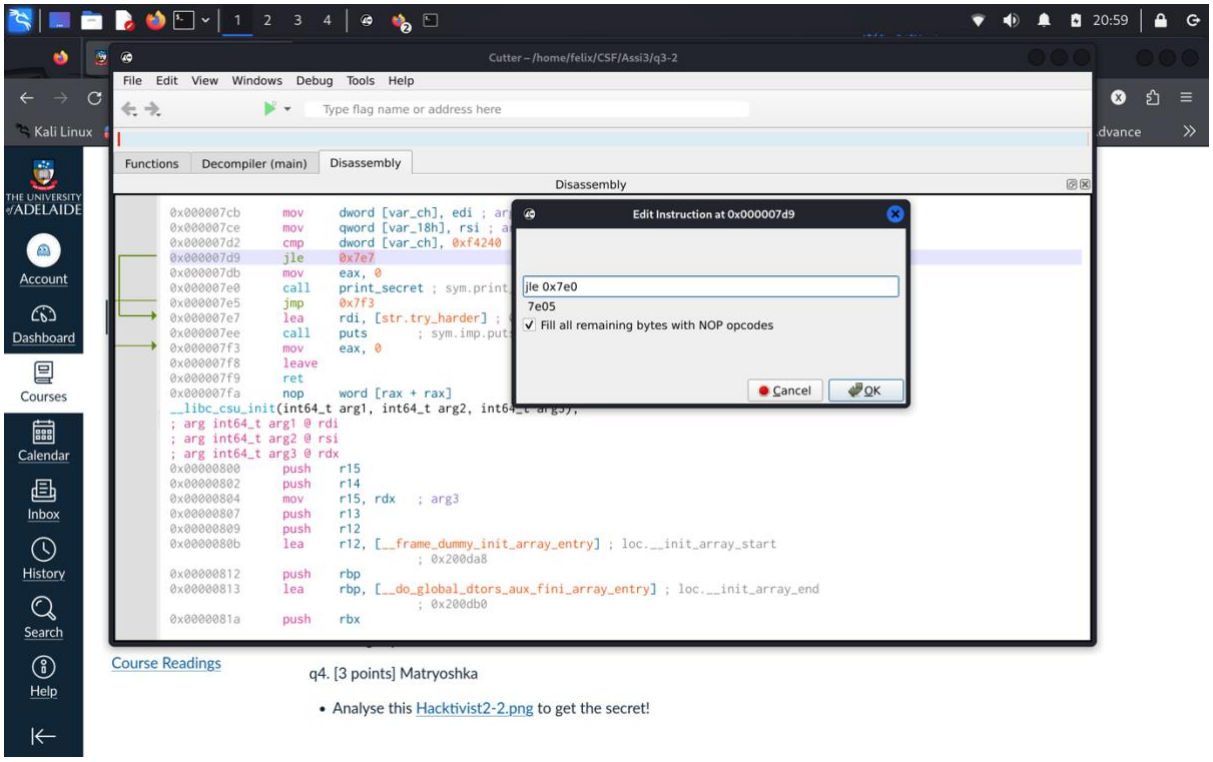
Finally, execute the q2 file again to get the secret:



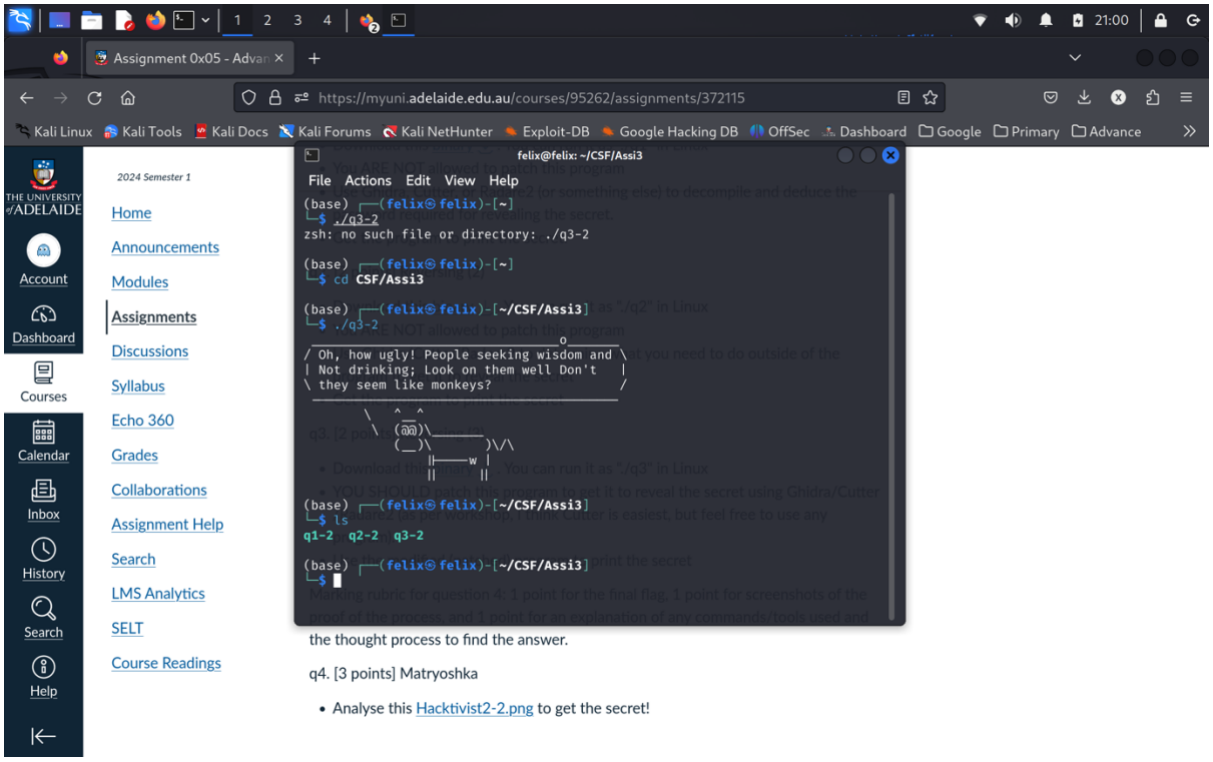
q3. [2 points] Reversing (3)

- Download this binary Download binary. You can run it as "./q3" in Linux
- YOU SHOULD **patch** this program to get it to reveal the secret using Ghidra/Cutter/Radare2 (as per workshop, I think Cutter is easiest, but feel free to use any program).
- Use the modified (**patched**) program to print the secret

Firstly, decompile the code using the write mode enable to patch it, see the disassembly view of the code to find that the address of the “print_secret” function is “7e0”, modify the original “jle” address from “7e7” to “7e0” to directly jump to the function even the variable less equal to 0xf4240:



After modified the address and quit cutter, the program can be directly run and show the secret:

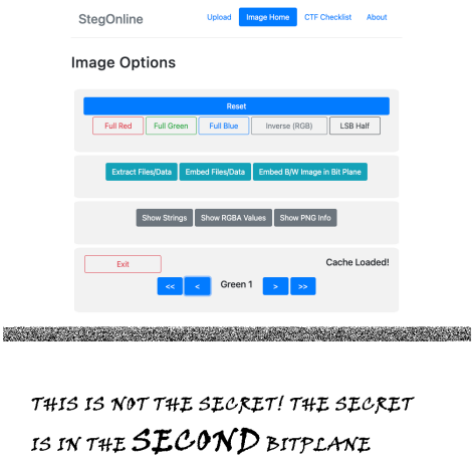


Marking rubric for question 4: 1 point for the final flag, 1 point for screenshots of the proof of the process, and 1 point for an explanation of any commands/tools used and the thought process to find the answer.

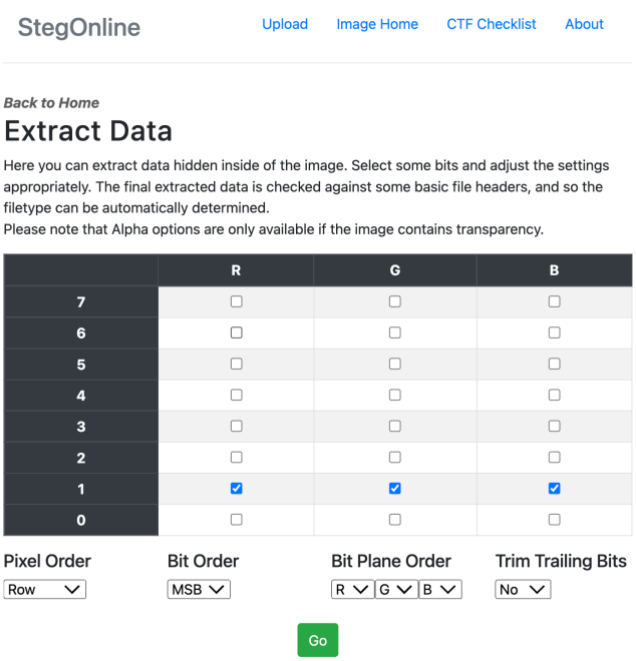
q4. [3 points] Matryoshka

- Analyse this Hacktivist2-2.png to get the secret!

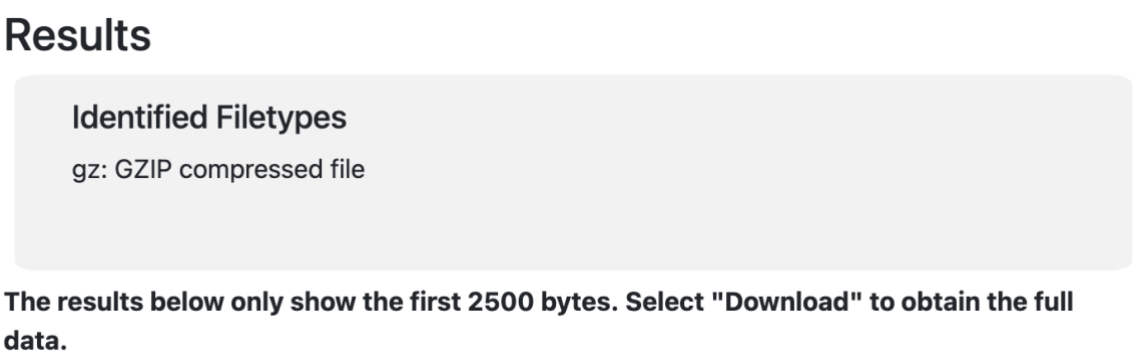
Firstly, use the website mentioned in the workshop and rotate through the “bitplanes”, got the hint:



Then use the “Extract Data” function and choose the second bitplane to extract data:



Then got a GZIP file:

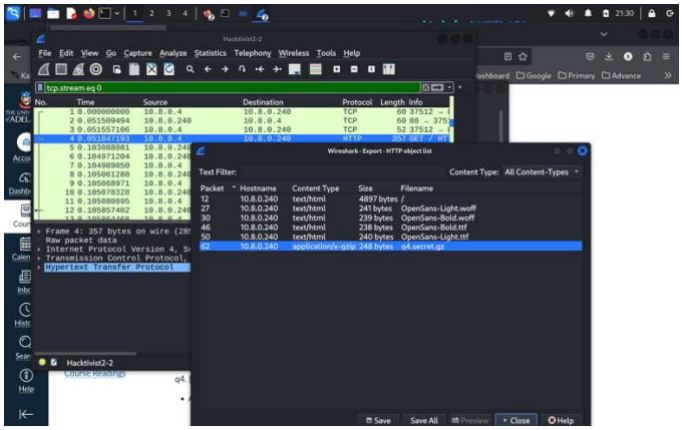


Download the GZIP file and extract it got a file named “Hacktivist2-2”, seem like a wireshark file because it shows “Dumpcap(Wireshark)” in the top lines of the file, so open it with wireshark, then find a file named “q4.secret.gz” in the HTTP stream:

```
$ cat Hacktivist2-2


?M<+????????6Intel(R) Core(TM) i7-8650U CPU @ 1.90GHz (with SSE4.2)Linux 5.5.0-k
ali2-amd64:Dumpcap (Wireshark) 3.2.3 (Git v3.2.3 packaged as 3.2.3-1)?Detun0
                                                                          L
inux 5.5.0-kali2-amd64D\c       ?/a?<<E<[?@@?


?P?/6???j?
i?}\\c  ?(s?<<E<@@%?
```



Finally, save the file, extract it, and got the secret: