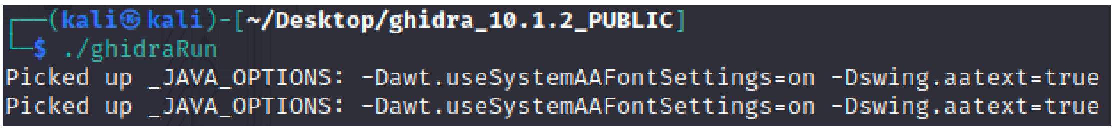
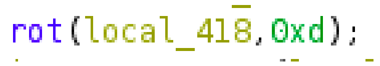
Q01

Step01: Analyze q1 with ghidra



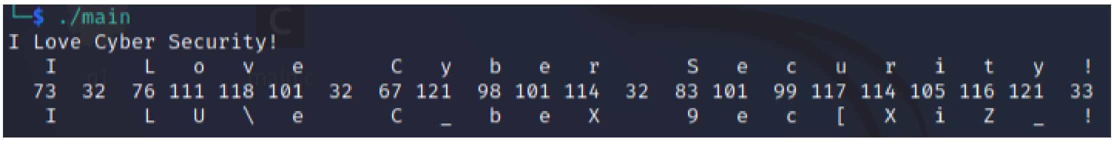
```
┌──(kali㉿kali)-[~/Desktop/ghidra_10.1.2_PUBLIC]
└─$ ./ghidraRun
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
```

Steop02: According to the report, the program “rot”our input



```
rot(local_418,0xd);
```

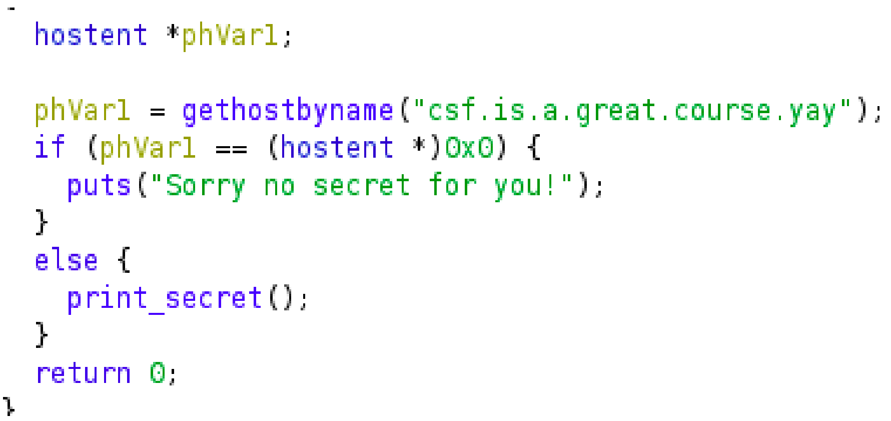
Step03: The password is “I Love Cyber Security!”But the output looks like this



```
└─$ ./main
I Love Cyber Security!
   I        L   o   v   e       C   y    b   e   r       S   e    c   u   r   i   t   y    !
  73  32   76 111 118 101  32  67 121   98 101 114  32  83 101   99 117 114 105 116 121   33
   I        L   U   \   e       C   _    b   e   X       9   e    c   [   X   i   Z   _    !
```

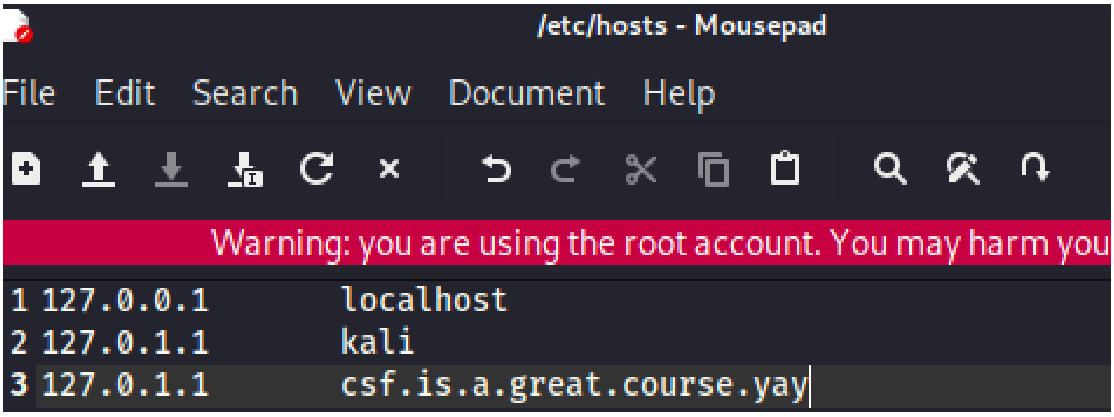
Q02

Step01: According to ghidre, the main function is to find the IP address of a host called “csf.is.a.great.course.yay”



```
  hostent *phVar1;

  phVar1 = gethostbyname("csf.is.a.great.course.yay");
  if (phVar1 == (hostent *)0x0) {
    puts("Sorry no secret for you!");
  }
  else {
    print_secret();
  }
  return 0;
}
```

Step02: Add the hostname to kali’s hosts

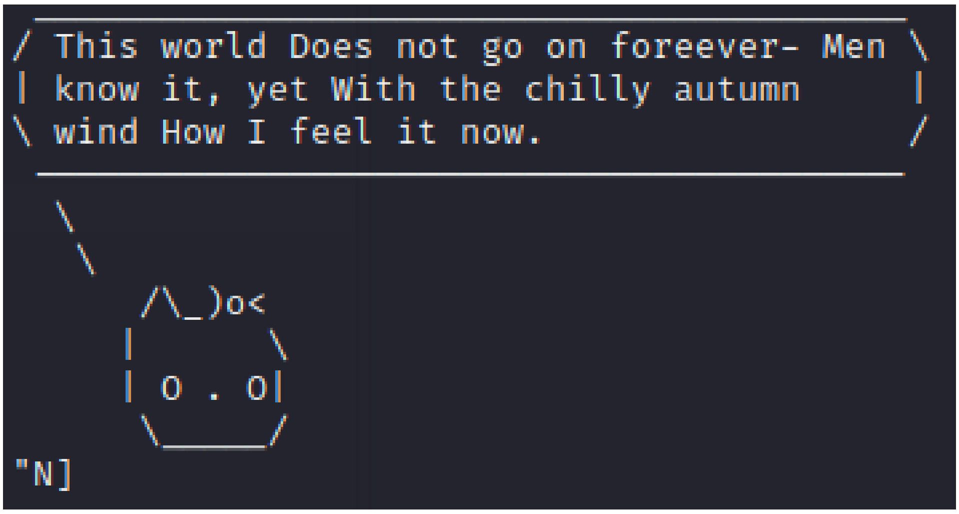


Warning: you are using the root account. You may harm you

```
1 127.0.0.1       localhost
2 127.0.1.1       kali
3 127.0.1.1       csf.is.a.great.course.yay
```

Step03: Run q2

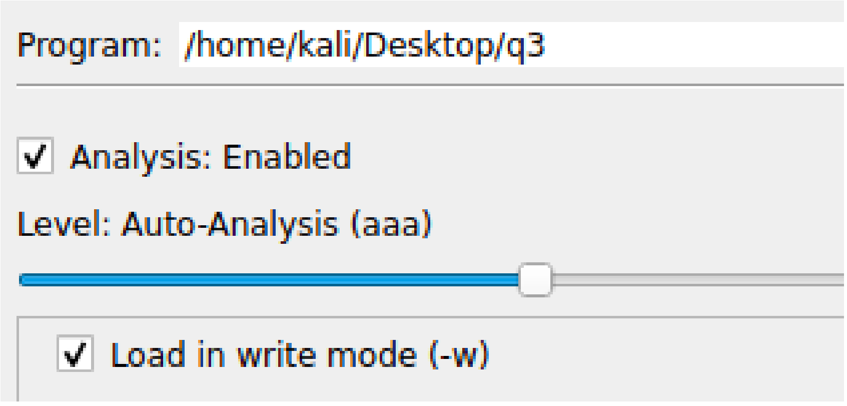
Answer:



```
 ________________________________________
/ This world Does not go on foreever- Men \
| know it, yet With the chilly autumn     |
\ wind How I feel it now.                 /
 ----------------------------------------
   \
    \
       /\_)o<
      |      \
      | O . O|
       \_____/
"N]
```

Q03

Step01: Open q3 with write mode



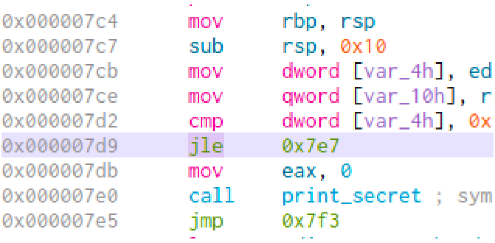
Program: /home/kali/Desktop/q3

☑ Analysis: Enabled

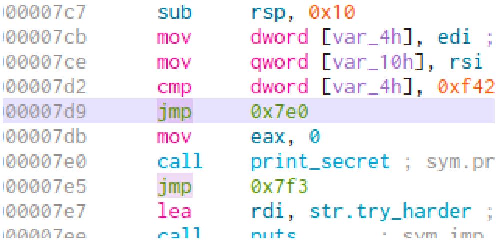
Level: Auto-Analysis (aaa)

☑ Load in write mode (-w)

Step02: Change jle 0x7e7 to jmp 0x7e0



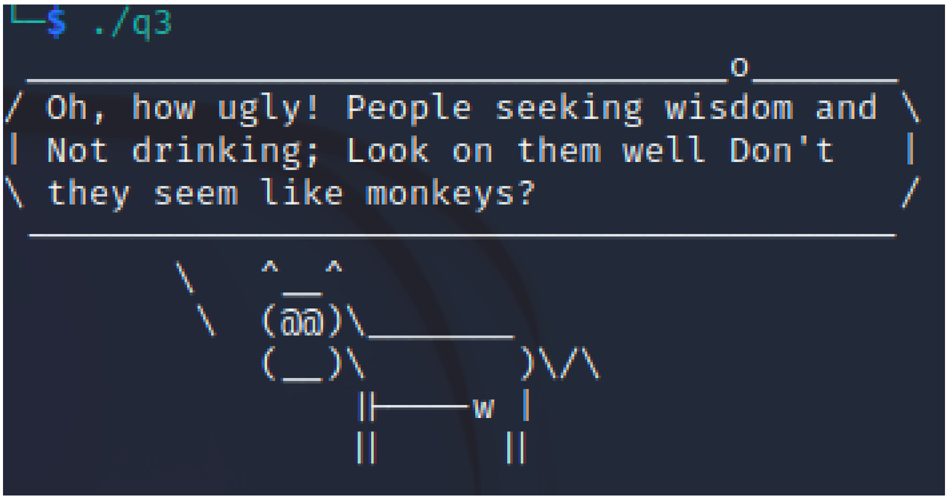
```
0x000007c4      mov     rbp, rsp
0x000007c7      sub     rsp, 0x10
0x000007cb      mov     dword [var_4h], ed
0x000007ce      mov     qword [var_10h], r
0x000007d2      cmp     dword [var_4h], 0x
0x000007d9      jle     0x7e7
0x000007db      mov     eax, 0
0x000007e0      call    print_secret ; sym
0x000007e5      jmp     0x7f3
```



```
00007c7      sub     rsp, 0x10
00007cb      mov     dword [var_4h], edi ;
00007ce      mov     qword [var_10h], rsi
00007d2      cmp     dword [var_4h], 0xf42
00007d9      jmp     0x7e0
00007db      mov     eax, 0
00007e0      call    print_secret ; sym.pr
00007e5      jmp     0x7f3
00007e7      lea     rdi, str.try_harder ;
00007ee      call    puts         ; sym.imp
```

Step03: Run q3

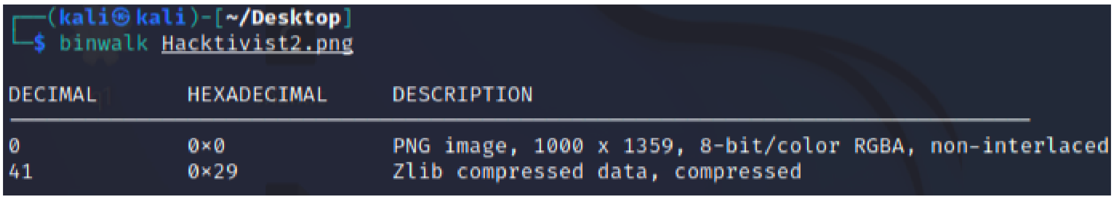
Answer:



```
└─$ ./q3
 _________________________________o________
/ Oh, how ugly! People seeking wisdom and \
| Not drinking; Look on them well Don't   |
\ they seem like monkeys?                 /
 ------------------------------------------
        \   ^__^
         \  (@@)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

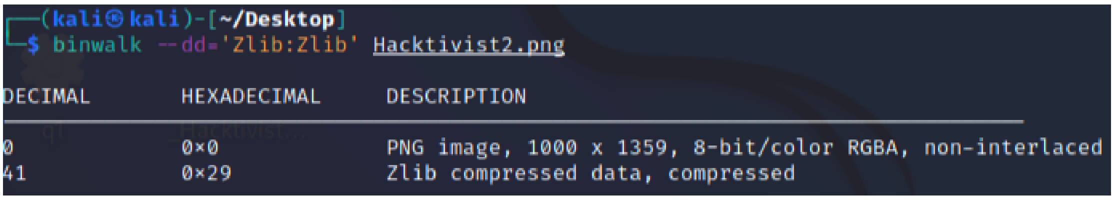
Q04

Step01: Use binwalk to analyze the file, found a Zlib



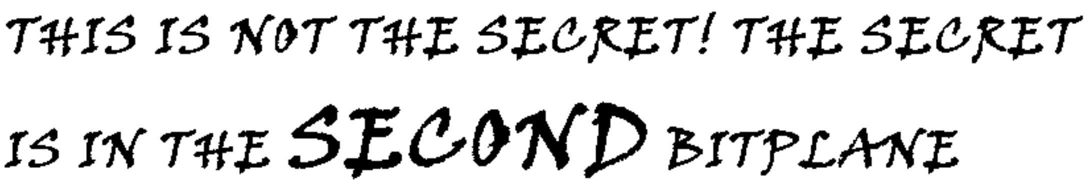
```
┌──(kali㉿kali)-[~/Desktop]
└─$ binwalk Hacktivist2.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 1000 x 1359, 8-bit/color RGBA, non-interlaced
41            0x29            Zlib compressed data, compressed
```

Step02: extract the Zlib



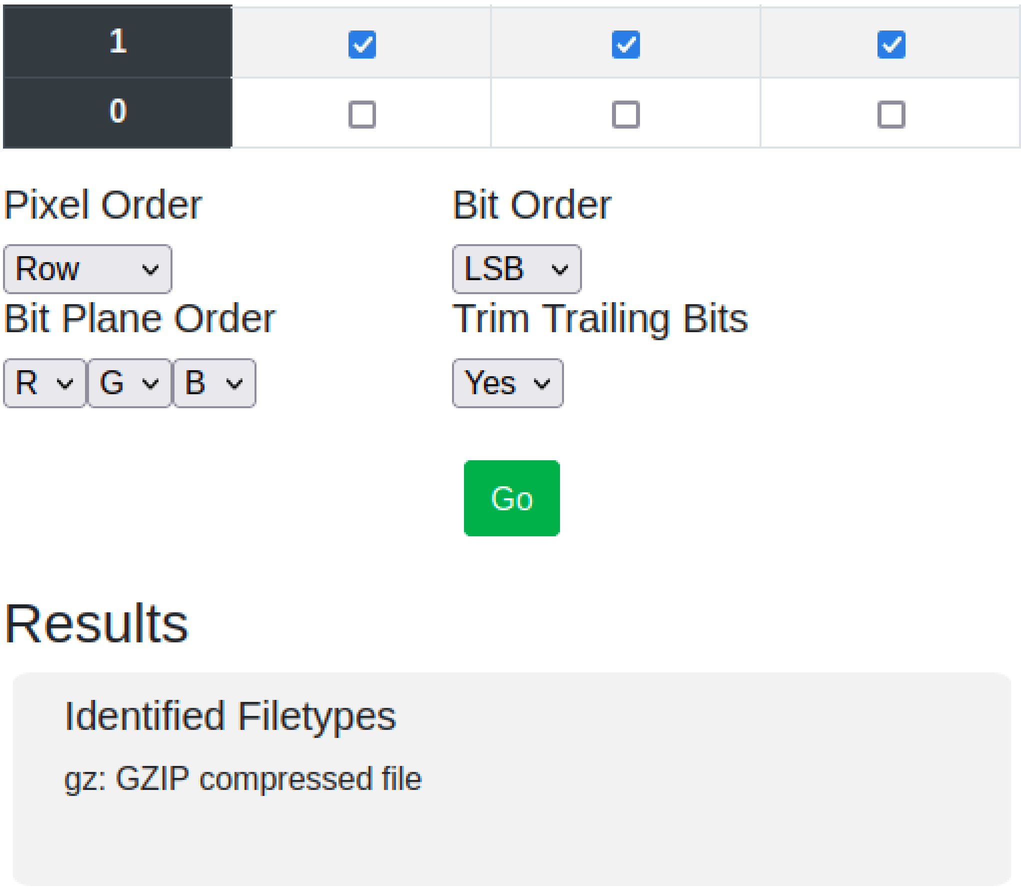
```
┌──(kali㉿kali)-[~/Desktop]
└─$ binwalk --dd='Zlib:Zlib' Hacktivist2.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 1000 x 1359, 8-bit/color RGBA, non-interlaced
41            0x29            Zlib compressed data, compressed
```

Step03: Browse the bit plane on stegonline, get a hint



THIS IS NOT THE SECRET! THE SECRET IS IN THE SECOND BITPLANE

Strp04: Select the second bitplane of this png, get a gzip



1	☑	☑	☑
0	□	□	□

Pixel Order: Row

Bit Order: LSB

Bit Plane Order: R G B

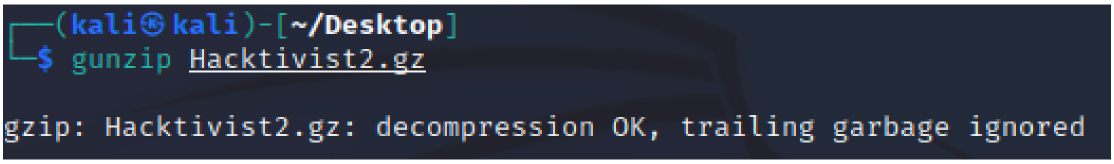
Trim Trailing Bits: Yes

Results

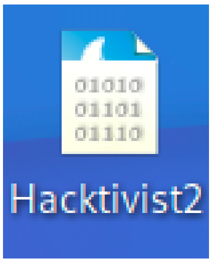
Identified Filetypes

gz: GZIP compressed file

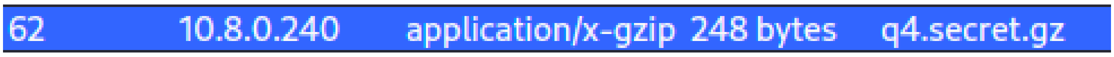
Step05: Extract a PCAPNG from the gzip



```
┌──(kali㉿kali)-[~/Desktop]
└─$ gunzip Hacktivist2.gz

gzip: Hacktivist2.gz: decompression OK, trailing garbage ignored
```

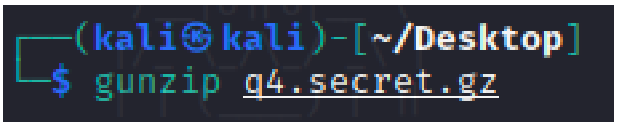


Step06: In Wireshark, export the q4.secret.gz



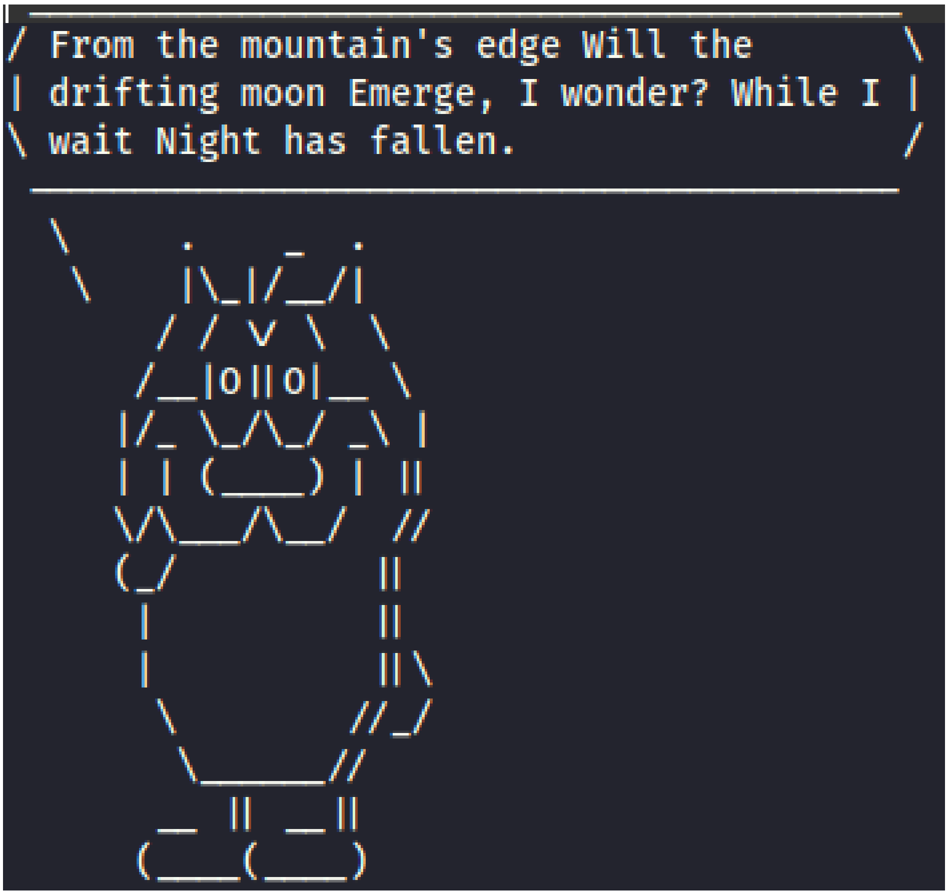
```
62          10.8.0.240      application/x-gzip  248 bytes     q4.secret.gz
```

Sttep07: Again, extract the secret from this gzip



```
┌──(kali㉿kali)-[~/Desktop]
└─$ gunzip q4.secret.gz
```

Answer:



```
 _________________________________________
/ From the mountain's edge Will the       \
| drifting moon Emerge, I wonder? While I |
\ wait Night has fallen.                  /
 -----------------------------------------
  \     .    _  .
   \    |\_|/__/|
       / /  v \  \
      /__|o||o|__ \
     |/_ \_/\_/ _\ |
     | | (____) | ||
     \/\___/\__/  //
     (_/         ||
      |          ||
      |          ||\
       \        //_/
        \______//
       __ || __||
      (____(____)
```