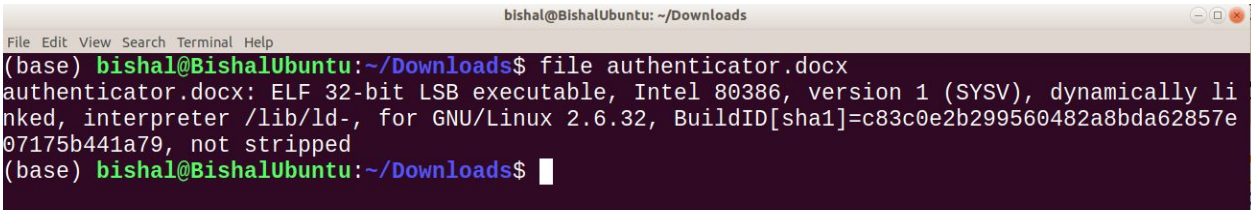
# Reverse Engineering

## Bishal Khadka

## Step 1:

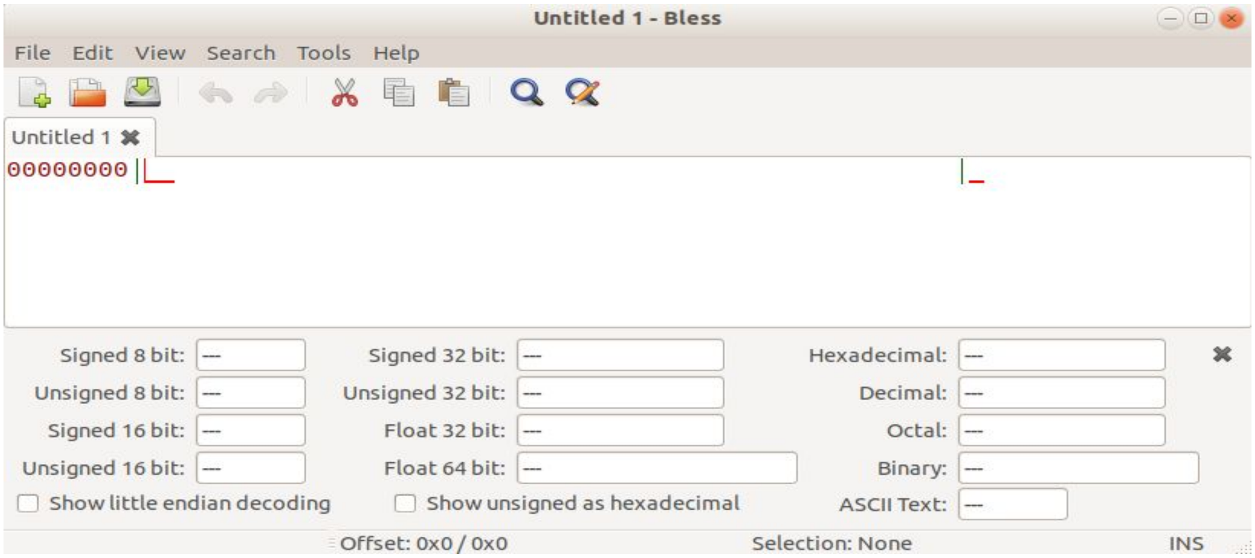
After Downloading authenticator.docx file, first check the type/format of the file. You can perform this by typing following command:
"**file file_name**": In our case: "**file authenticator.docs**"
Basically file command gives you the overview of the type of the file you will be running. For example, for text files it will print ascii text. For our authenticator.docs file, the following text was printed when typing the above command.


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

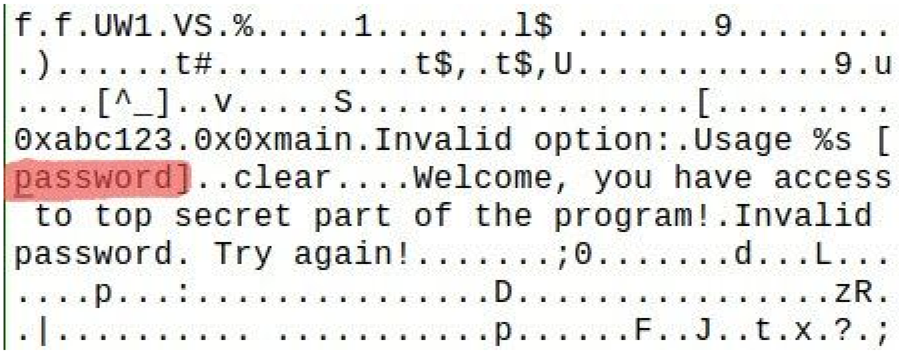
```
(base) bishal@BishalUbuntu:~/Downloads$ file authenticator.docx
authenticator.docx: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically li
nked, interpreter /lib/ld-, for GNU/Linux 2.6.32, BuildID[sha1]=c83c0e2b299560482a8bda62857e
07175b441a79, not stripped
(base) bishal@BishalUbuntu:~/Downloads$
```

## Step 2:

In my case, I used a hex editor called Bless Hex Editor for manipulation of the fundamental binary data. Hex editors are mainly used to see or edit the raw and exact contents of a file and may be used to correct data corrupted by a system or application program. Bless hex editor looks something like the following:

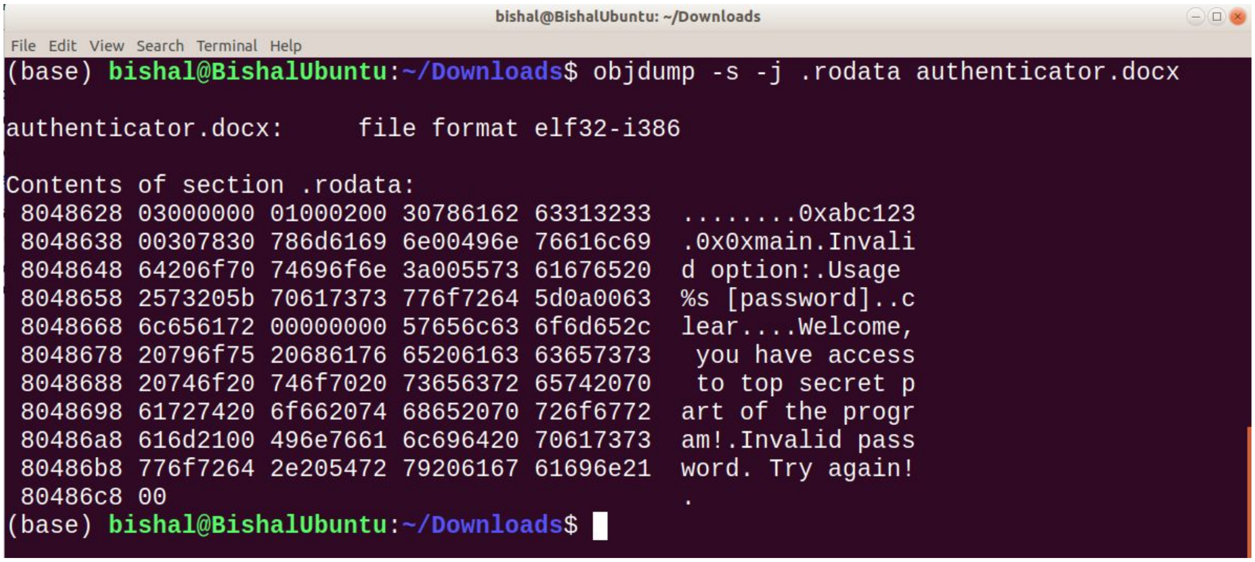


- Open the authenticator.docx file in bless hex editor: This can be accomplished without using hex editor rather by typing the command "**hexdump -C filename**"
- Look for the string like password in the right hand side of the box which displays the ascii character of the value on the left hand side box.



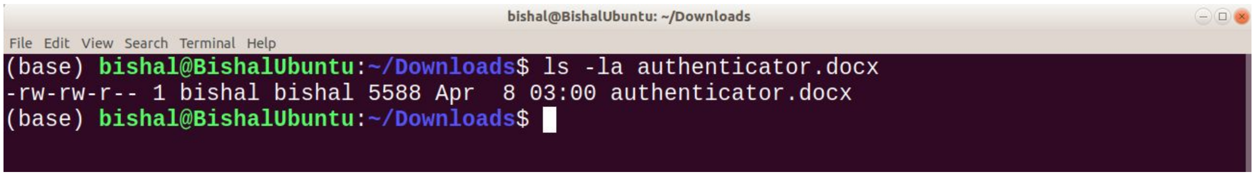
```
f.f.UW1.VS.%.....1.......l$ .......9........
.)......t#.........t$,.t$,U............9.u
....[^_]..v.....S.................[.........
0xabc123.0x0xmain.Invalid option:.Usage %s [
password]..clear....Welcome, you have access
 to top secret part of the program!.Invalid
password. Try again!.......;0.......d...L...
....p...:...............D................zR.
.|..........  ..........p......F..J..t.x.?.;
```

- Look to see if you find any useful information in the above displayed area. Ahh, the only information I find worth checking is ***0xabc123*** which might be the possible password for the vault.
- You can also type "**objdump -s -j .rodata filename**" inorder to see the read only data or literal strings contained in the file. This way is slightly easier to find the password.


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

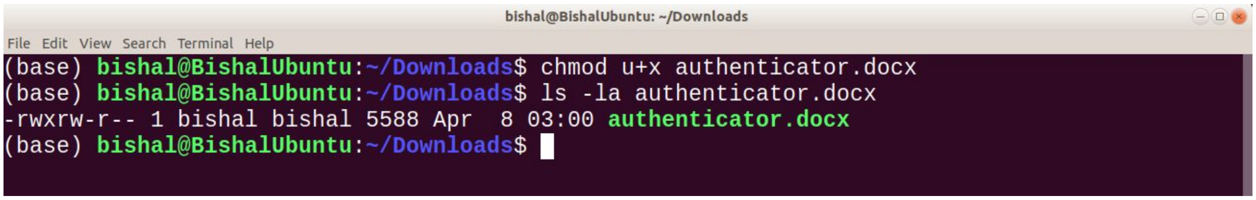
```
(base) bishal@BishalUbuntu:~/Downloads$ objdump -s -j .rodata authenticator.docx

authenticator.docx:     file format elf32-i386

Contents of section .rodata:
 8048628 03000000 01000200 30786162 63313233  ........0xabc123
 8048638 00307830 786d6169 6e00496e 76616c69  .0x0xmain.Invali
 8048648 64206f70 74696f6e 3a005573 61676520  d option:.Usage 
 8048658 2573205b 70617373 776f7264 5d0a0063  %s [password]..c
 8048668 6c656172 00000000 57656c63 6f6d652c  lear....Welcome,
 8048678 20796f75 20686176 65206163 63657373   you have access
 8048688 20746f20 746f7020 73656372 65742070   to top secret p
 8048698 61727420 6f662074 68652070 726f6772  art of the progr
 80486a8 616d2100 496e7661 6c696420 70617373  am!.Invalid pass
 80486b8 776f7264 2e205472 79206167 61696e21  word. Try again!
 80486c8 00                                   .
(base) bishal@BishalUbuntu:~/Downloads$ 
```

- The challenge is to find a way to run authenticator.docx file and enter the password. To do that first make the file authenticator.docx executable. Check the original flag in a file by typing “**ls -la filename**”


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

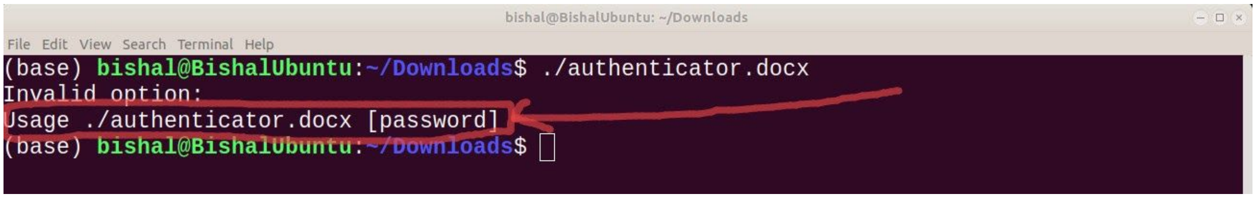
```
(base) bishal@BishalUbuntu:~/Downloads$ ls -la authenticator.docx
-rw-rw-r-- 1 bishal bishal 5588 Apr  8 03:00 authenticator.docx
(base) bishal@BishalUbuntu:~/Downloads$ 
```

- Now, make it executable by changing the flag of the file by giving the user permission to execute the file by typing “**chmod 764 authenticator.docx**” or “**chmod u+x authenticator.docs**” for the authenticator.docs file. After typing any of those command/s, the file should be reflected in green color as follows.


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

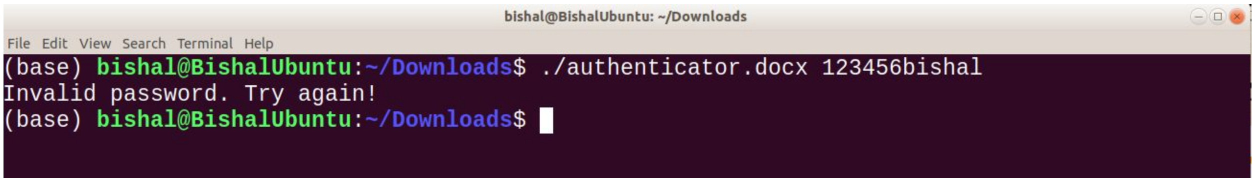
```
(base) bishal@BishalUbuntu:~/Downloads$ chmod u+x authenticator.docx
(base) bishal@BishalUbuntu:~/Downloads$ ls -la authenticator.docx
-rwxrw-r-- 1 bishal bishal 5588 Apr  8 03:00 authenticator.docx
(base) bishal@BishalUbuntu:~/Downloads$ 
```

- Since our file is now executable, try running it by typing “**./authenticator.docx**”. It will provide us a little hint if any other argument should be passed while executing this file. In the following screenshot, glance at the usage string to know how the file should be executed.


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

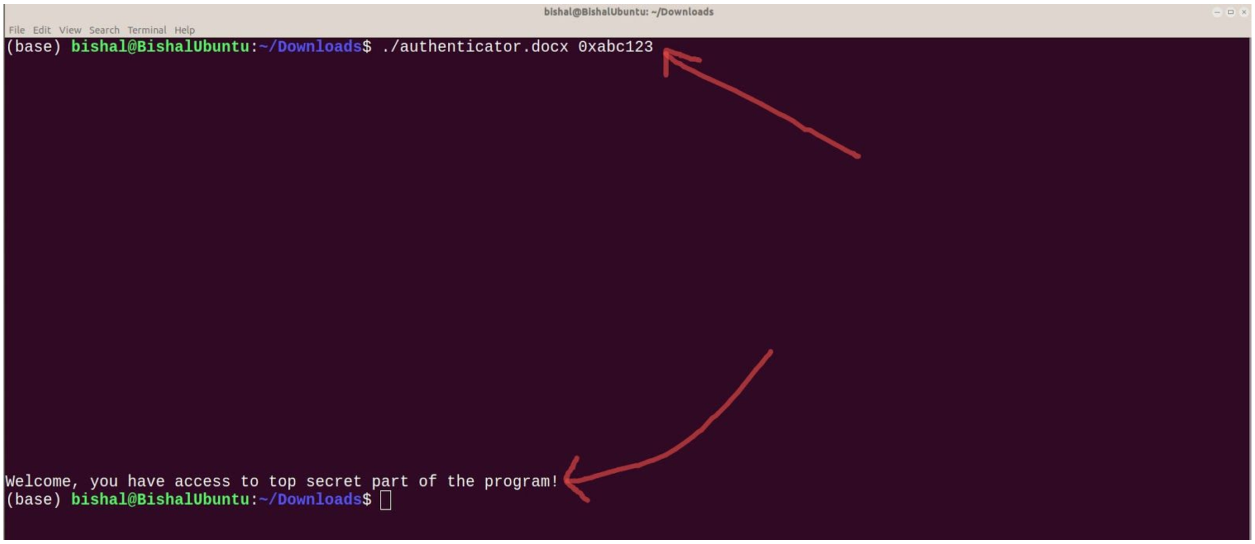
```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx
Invalid option:
Usage ./authenticator.docx [password]
(base) bishal@BishalUbuntu:~/Downloads$ 
```

- The above screenshot says that while running ./authenticator.docx file, some password should be passed with it as a second argument vector. For example, run this file by: **./authenticator somePassword**, where ./authenticator is the first argument and somePassword is the second argument in the command line.
- Since we have already predicted our password using the hex editor, it is now time to bring that predicted password into operation. Before that let’s try with the incorrect password.


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx 123456bishal
Invalid password. Try again!
(base) bishal@BishalUbuntu:~/Downloads$ 
```

- Finally, now let’s deploy the correct password (0xabc123).


bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

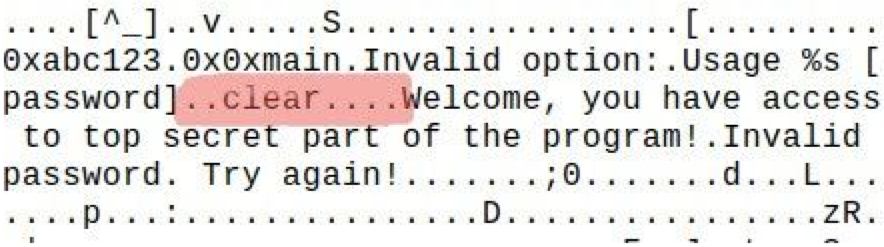
```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx 0xabc123









Welcome, you have access to top secret part of the program!
(base) bishal@BishalUbuntu:~/Downloads$ 
```

## Step 3:

Modifying the binary so that it executes /bin/sh when successfully authenticated.

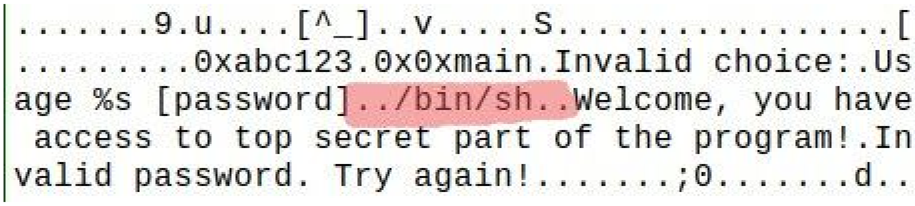
- First, open up the hex editor “**hexedit filename**” or bless hex editor.

- Since system call makes the system vulnerable as if the application ever runs as a privileged user, all a hacker has to do is put their own program in the system call and run it.
- Find any system call in the binary file. In our case, I saw the system call called "**clear**" which clears the screen and gets us to the main vault.



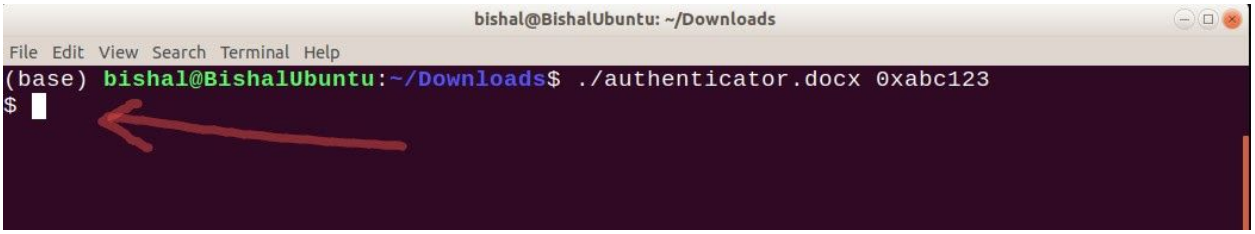
```
....[^_]..v.....S.................[.........
0xabc123.0x0xmain.Invalid option:.Usage %s [
password]..clear....Welcome, you have access
 to top secret part of the program!.Invalid
password. Try again!.......;0.......d...L...
....p...:...............D................zR.
```

- Play with the hex editor if you can find a way to modify it. Most of the time, I got segmentation fault as I changed to modify the byte of the binary file. However, after a couple of tries without modifying the byte, I was able to run the /bin/sh shellcode. If you want to execute your own shell code, the number of bytes of your file has to match the number of bytes of the system call.



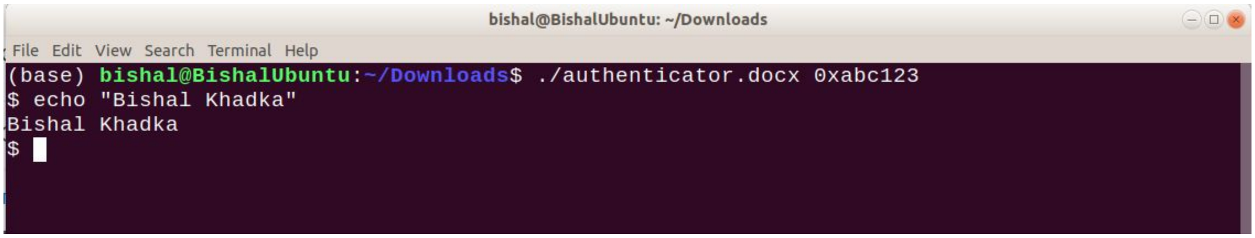
```
.......9.u....[^_]..v.....S.................[
.........0xabc123.0x0xmain.Invalid choice:.Us
age %s [password]../bin/sh..Welcome, you have
 access to top secret part of the program!.In
valid password. Try again!.......;0.......d..
```

- After running the binary file "./authenticator.docx 0xabc123", I was able to execute the /bin/sh shellcode.



bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

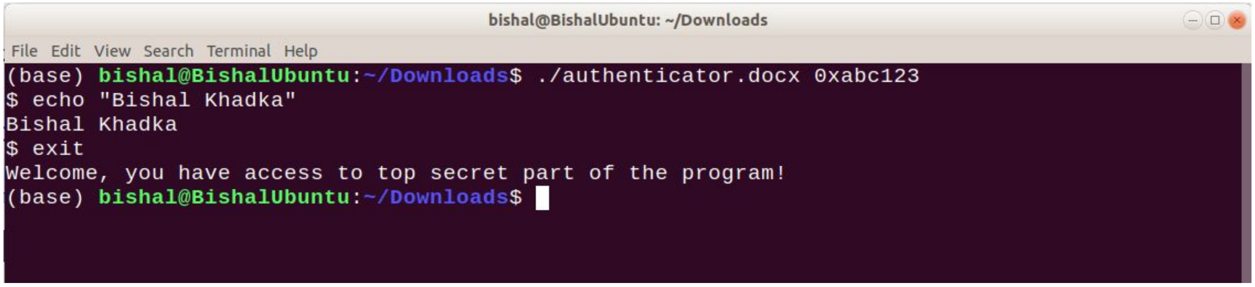
```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx 0xabc123
$ 
```



bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx 0xabc123
$ echo "Bishal Khadka"
Bishal Khadka
$ 
```

- After hitting exit in the shellcode, I was in the vault.



bishal@BishalUbuntu: ~/Downloads
File Edit View Search Terminal Help

```
(base) bishal@BishalUbuntu:~/Downloads$ ./authenticator.docx 0xabc123
$ echo "Bishal Khadka"
Bishal Khadka
$ exit
Welcome, you have access to top secret part of the program!
(base) bishal@BishalUbuntu:~/Downloads$ 
```