Overthewire bandit

***Level 0***

The host to which we need to connect is bandit.labs.overthewire.org on port 2220.username is bandit0 and the password is bandit0. The password for the next level was stored in a file called readme , in the home directory.

Here, you need to read the content of the readme file with the command cat.

Cat command is used to read data from the file and give its content as output.

***Level1***

The password for the next level is stored in a file called " - " located in the home directory

As '-'  means reading from or to stdin in a shell, its necessary to specify a path to read the file.

***Level2***

The password for the next level is stored in a file called "spaces in this filename "located in the home directory.Here you must make use of \ or " " while reading data using spaces in this filename

***Level3***

The password for the next level is stored in file in the "inhere" directory.

Here ,a hidden file is any file that begins with a “.”. When a file is hidden it can not been seen with the bare ls command in order to see hidden files using the ls command ,add the -a switch.

***Level4***

The password for the next level is stored in the inhere directory.

***Level5***

The password for the next level is stored in a file in the inhere directory .To do this ,I used  the find command to look for a specific file. Here, i used the -readable ,! -executable and -size 1033c parameters to find the file .

***Level6***

The password for the next level is stored somewhere on the server has group bandit6, user is bandit7. Here ,initially I found it hard to understand but later i used command grep to Search for a specific string in an output.

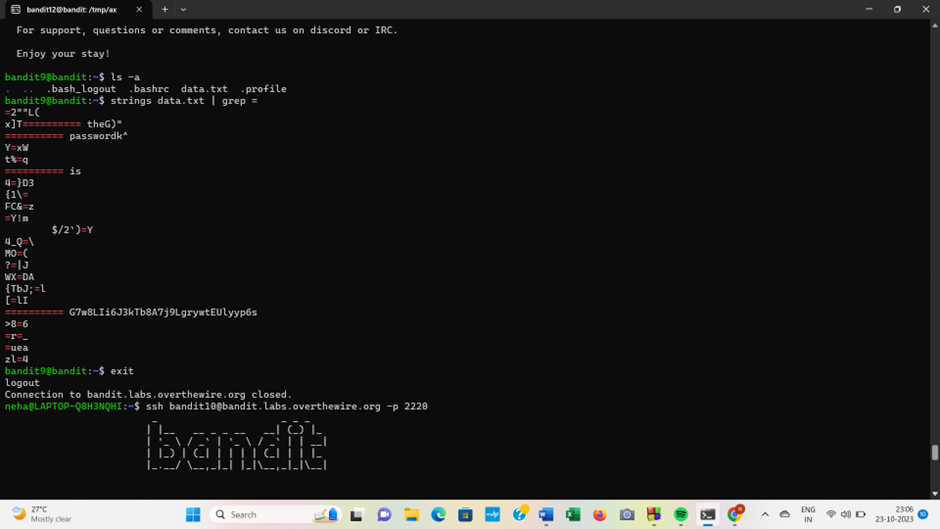
***Level7***

The password for the next level is stored in the file data.txt .Here we use the -exec argument of find with the grep command to find the file containing the word millionth.

***Level8***

For the next level the password is stored in the file data.txt and I found that it has the only line of text that occurs only once.Initially I used sort to sort alphabetically the data in the data.txt file then, we use uniq to count the number of occurances and find that one line of text that occurs only once

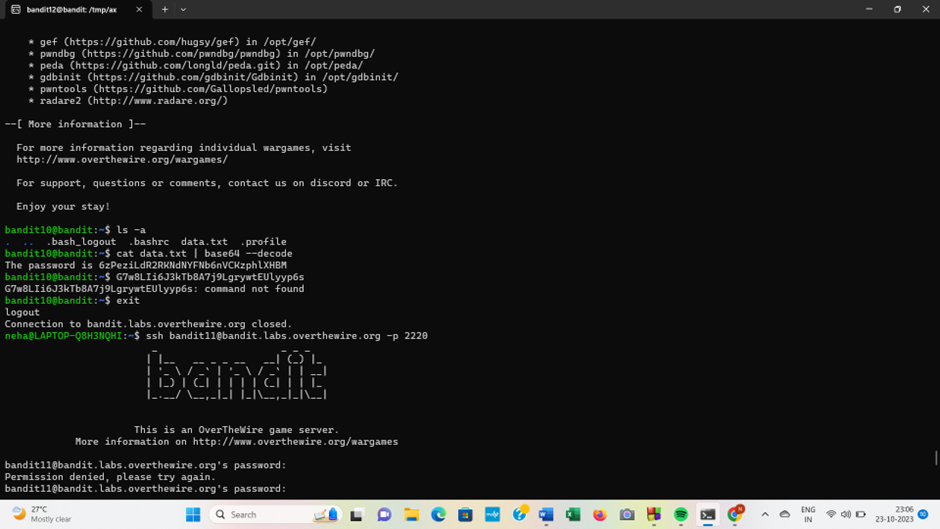
***Level9***

The password for the next level was stored in the file data.txt in a string, begins with several ‘=’ characters.The strings command helped me to find the string then grep found the string beginning with several ‘=’ characters.

***Level10***

The password for the next level is stored in the file data.txt which contains base64 encoded data.

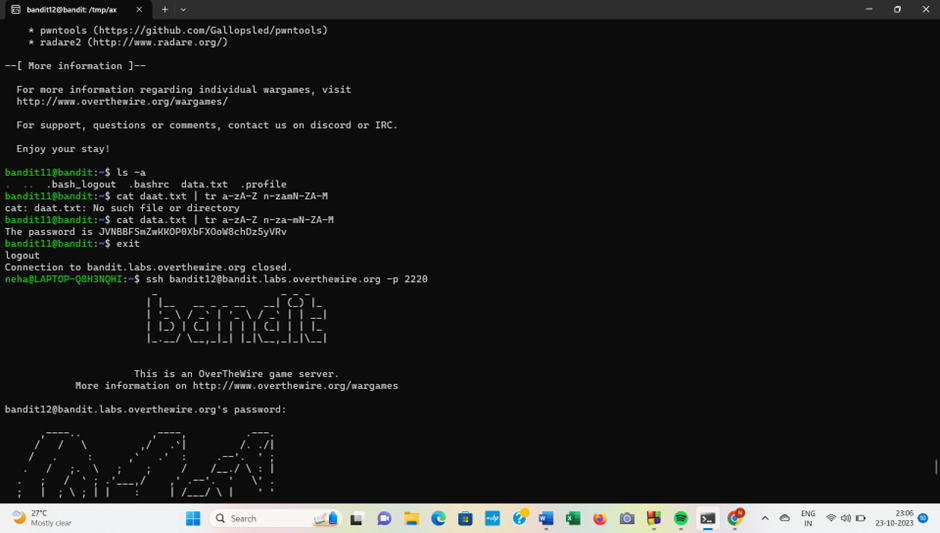
I understood what base64 was and then I read the data.txt and redirected the output to the base64 command. Also I used  -d argument to decode the string.“Base64 is a binary to a text encoding scheme that represents binary data in an American Standard Code for Information Interchange (ASCII) string format.”



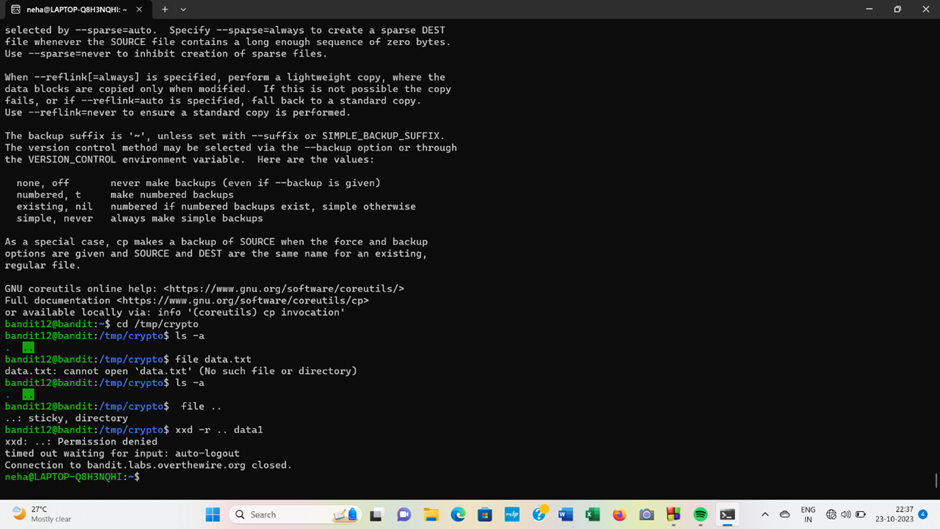
***Level11***

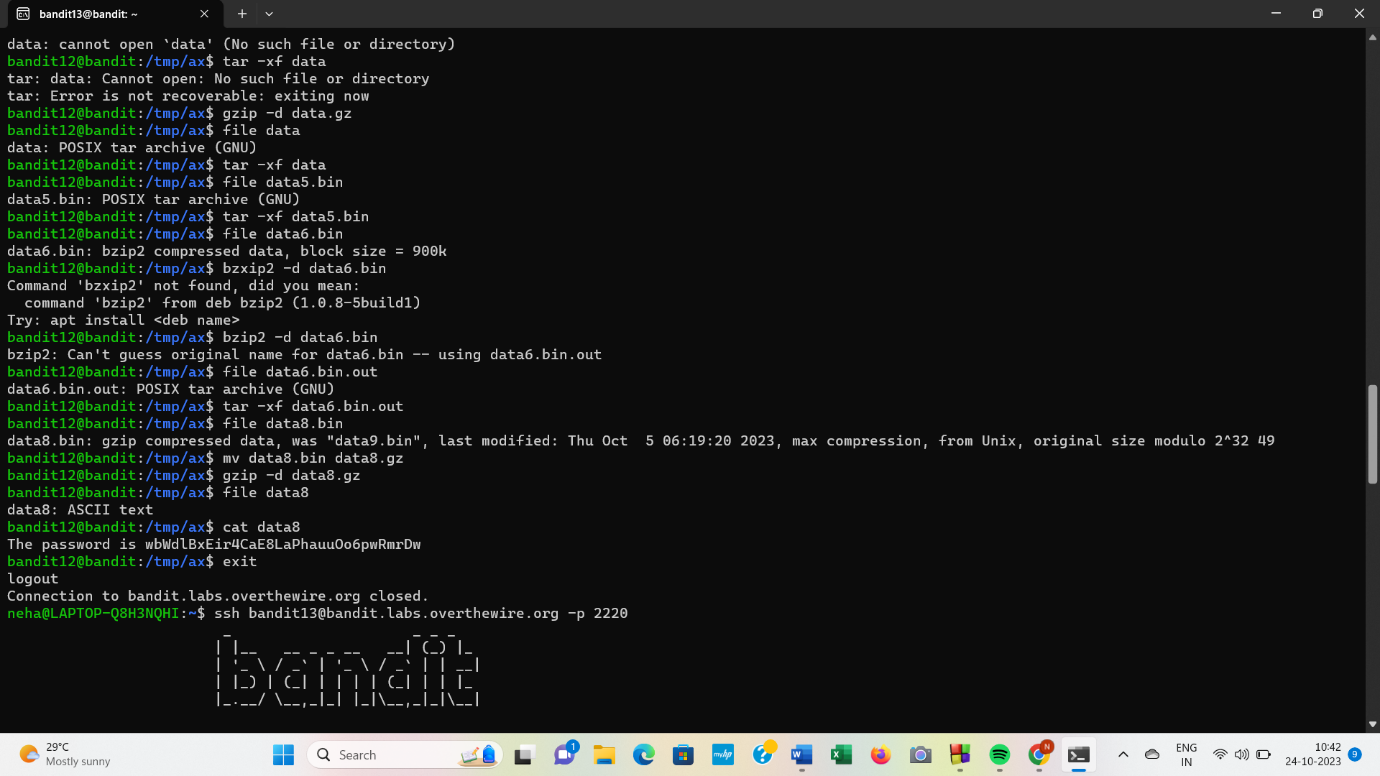
The next level is cleared by inserting password stored in the file data.txt. Here , all lowercase (a-z) and uppercase (A-Z) letters have been rotated by 13 positions. Initially ,I was not aware about it then I googled what is ROT13 and I used  tr command to translate the first set of characters 'A-Za-z 'to 'N-ZA-Mn-za-m '.

tr command is used to translate or delete characters from standard input and writes the result to standard output .



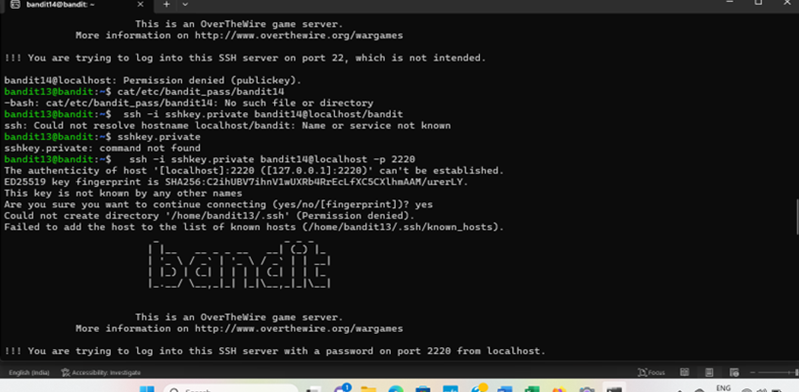
***level12***

This level was not that easy to be cleared ; First we have to create a folder under the tmp directory to do this we run “mkdir /tmp/idk” Once the folder is created ,I copied the data.txt file to the folder with “cp data.txt /tmp/idk”. In order to decypher the hexdump of the file I ran xxd -r data.txt > bandit which reverses the hexdump and saves it into a file named bandit. I came across the hint that the file was compressed multiple times so I ran "file bandit" to see if it can determine the compression method used and found that it was compressed with gzip.Further I ran certain commands and finally found the password .I also commited many mistakes as shown below



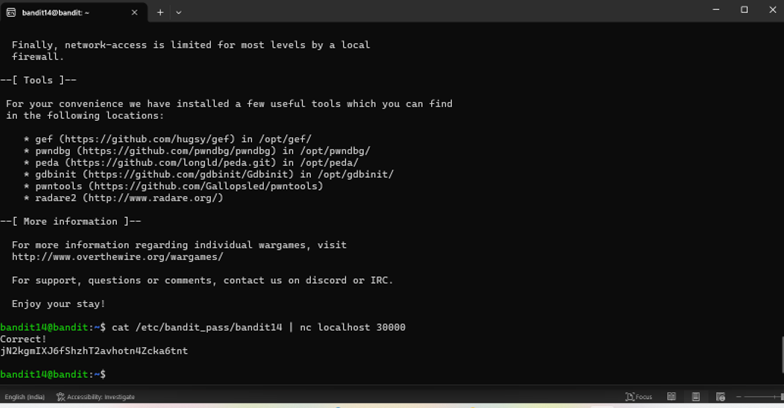
***Level 13***

The password for the next level is stored in /etc/bandit\_pass/bandit14 and can only be read by user bandit14.When I logged into the machine and I ran “ls”I saw a file named sshkey.private ;which must be the SSH private key . I ran “ssh -i sshkey.private bandit14@localhost”. The -i means that I am using an identity file to login to bandit14 on the server since all of the bandit users are on the same machine. After I logged into bandit14 ,I ran “cd /etc/bandit\_pass/ “,as specified on the hint and then I read file bandit14 which gave me the next password.



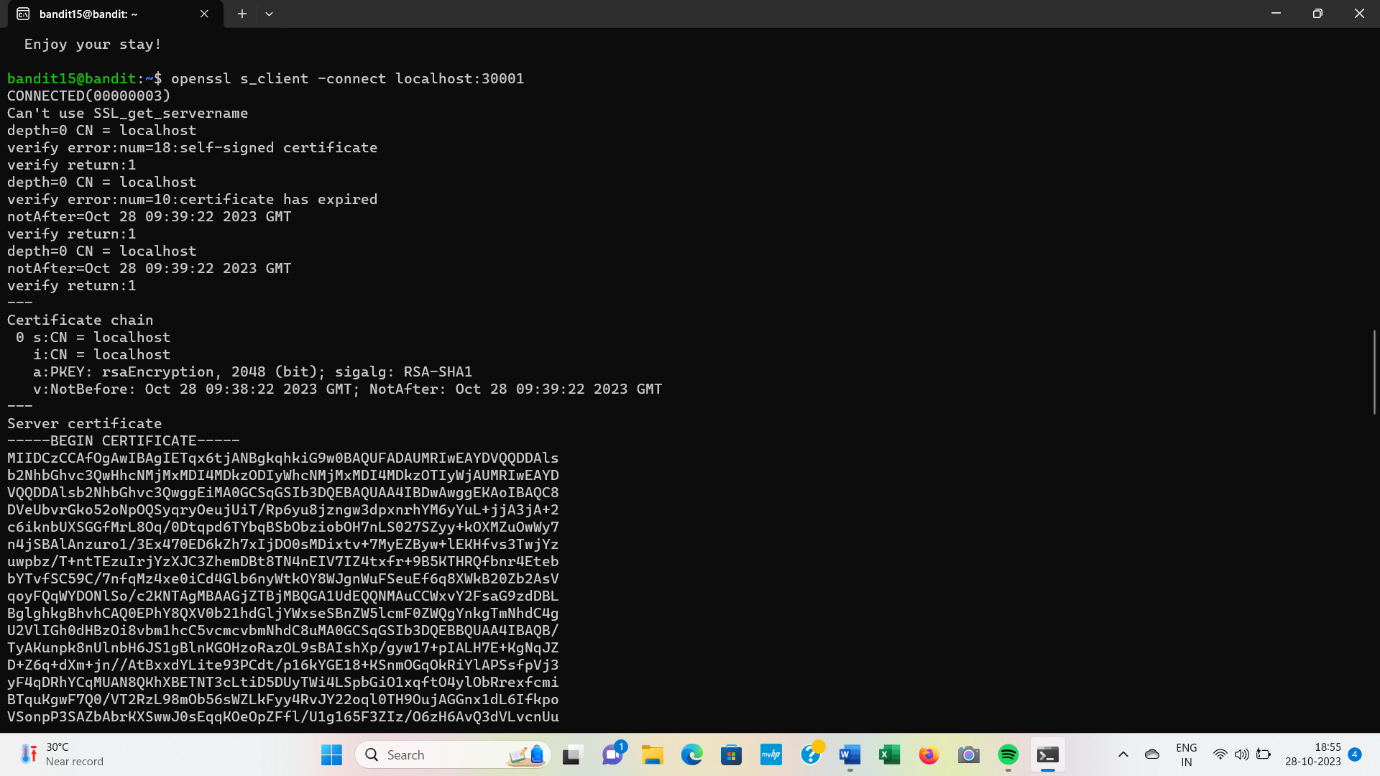
***Level14***

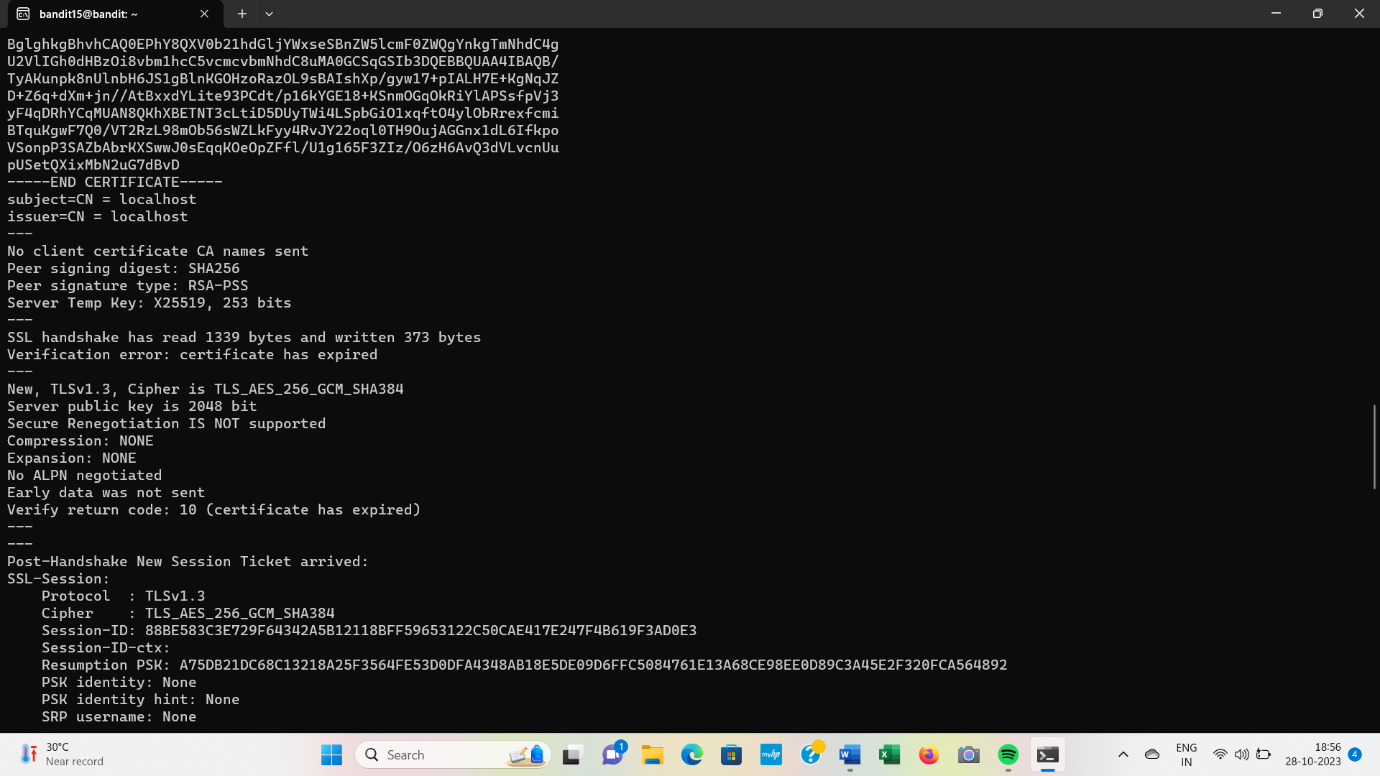
The password for the next level can be retrieved by submitting the password of the current level to port 30000 on localhost.We have to connect to port 30000 on localhost and we have to send a string containing the current password. To do this I ran “nc localhost 30000” and once I saw that I was connected ,I pasted the password

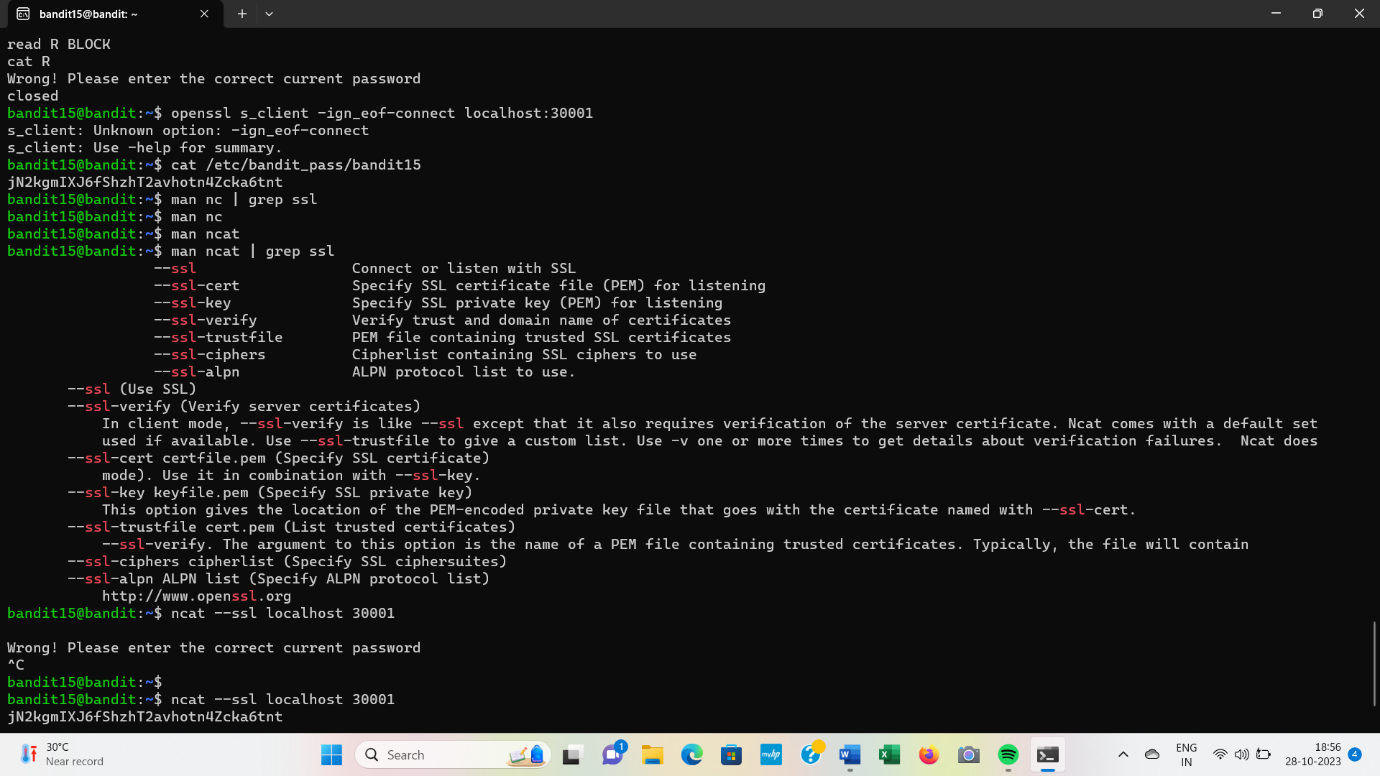
bandit14 and then I received the following output

***Level15***

The password for the next level can be retrieved by submitting the password of the current level to port 30001 on localhost using SSL encryption.I tried connecting to the port 30001 with the command which was working and I was able to type in the password of bandit15. After I typed in the password I received some irrelevant messages.They were the error message the hint was talking about. After reading the man page of openssl I tried looking up the pattern -ign\_eof by hitting the / key and typing the string. However I was unable to find anything about it. I then tried to see what options are available for s\_client by running openssl s\_client -h which was not a valid command but I did get all of the options for s\_client. I was then able to find the -ign\_eof option the hint was talking about. As explained by the manual "-ign\_eof" ignores the input eof. When we run the command below and I typed the password for bandit15 I received the new password.



t

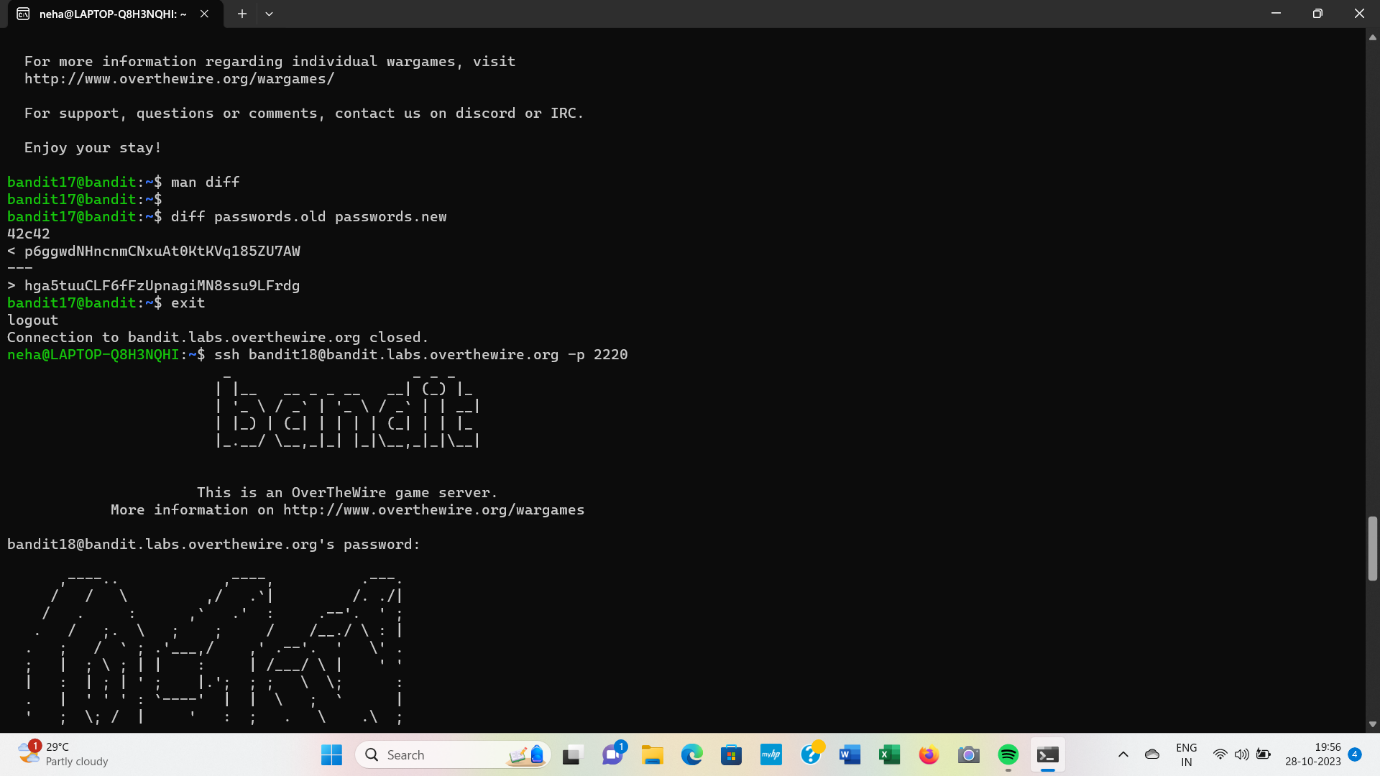
I *have also displayed the problems I faced while clearing this level. But finally I realized it and resolved them.*

***level 16***

To Jump into next level we must submit the password of the current level to a port on localhost in the range 31000 to 32000. There is only 1 server that will give the next credentials needed, the others will simply send back to you whatever you send to it.

As explained by the hint ,I found which host is up and running SSL. To do this, I ran a nmap that helped me check every port from 31000 to 32000 and check what services were running on that port. To do this I ran some the commands and finally arrived to the password for the next level.

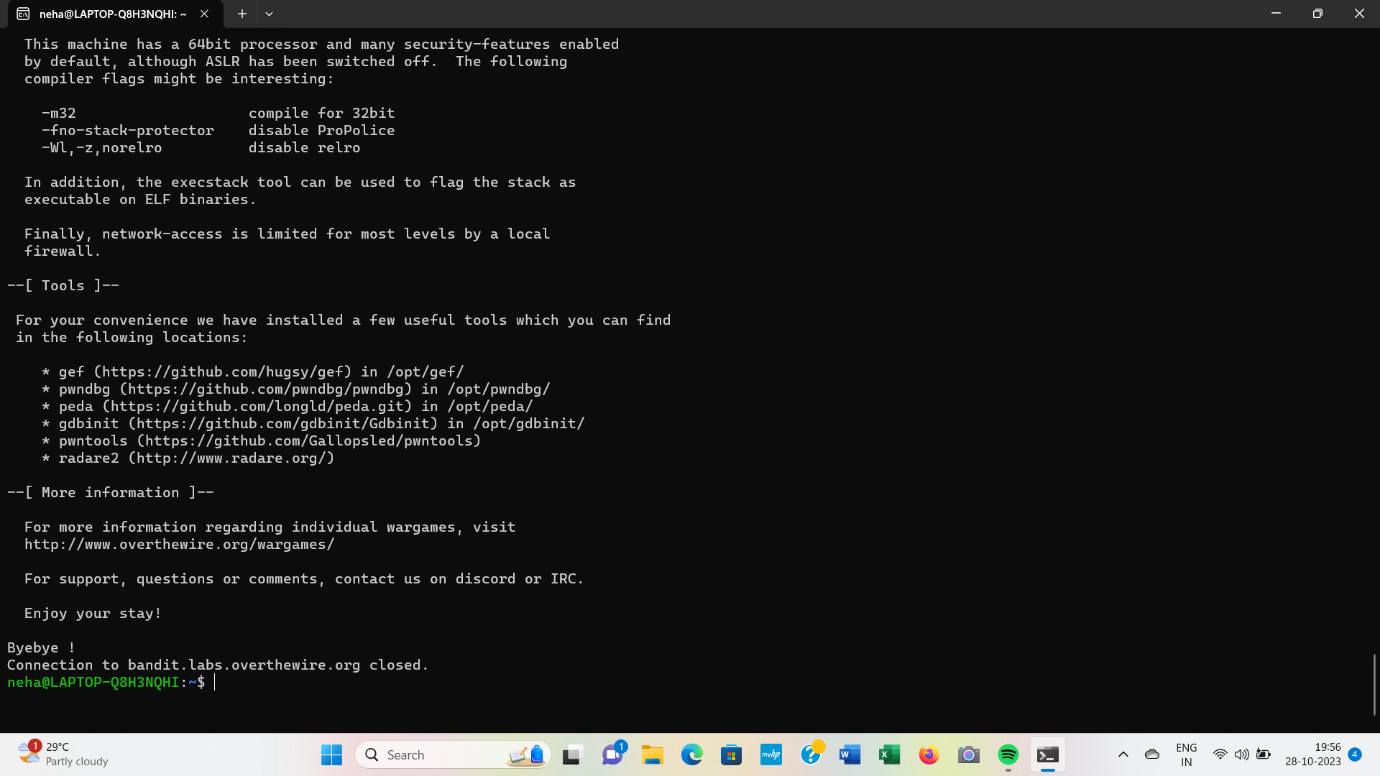
***Level17***

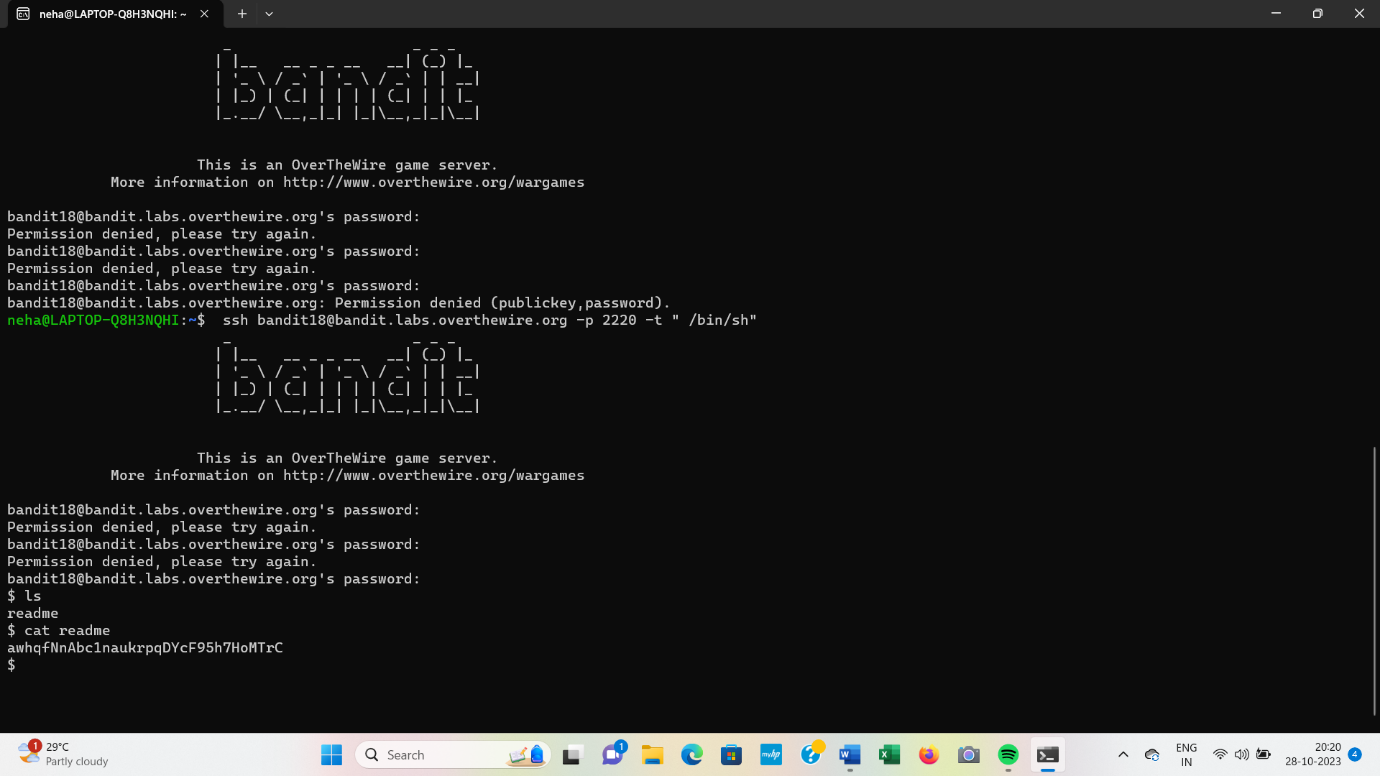
In this level ,there are 2 files in the home directory ;passwords.old and passwords.new. The password for the next level is in passwords.new .To do this ,I used diff command

***Level 18***

In this level; I tried to login with both password but only the 1st one worked and just like the hint said- I received the "Byebye!" output and I was kicked out. To retrive password for next level-

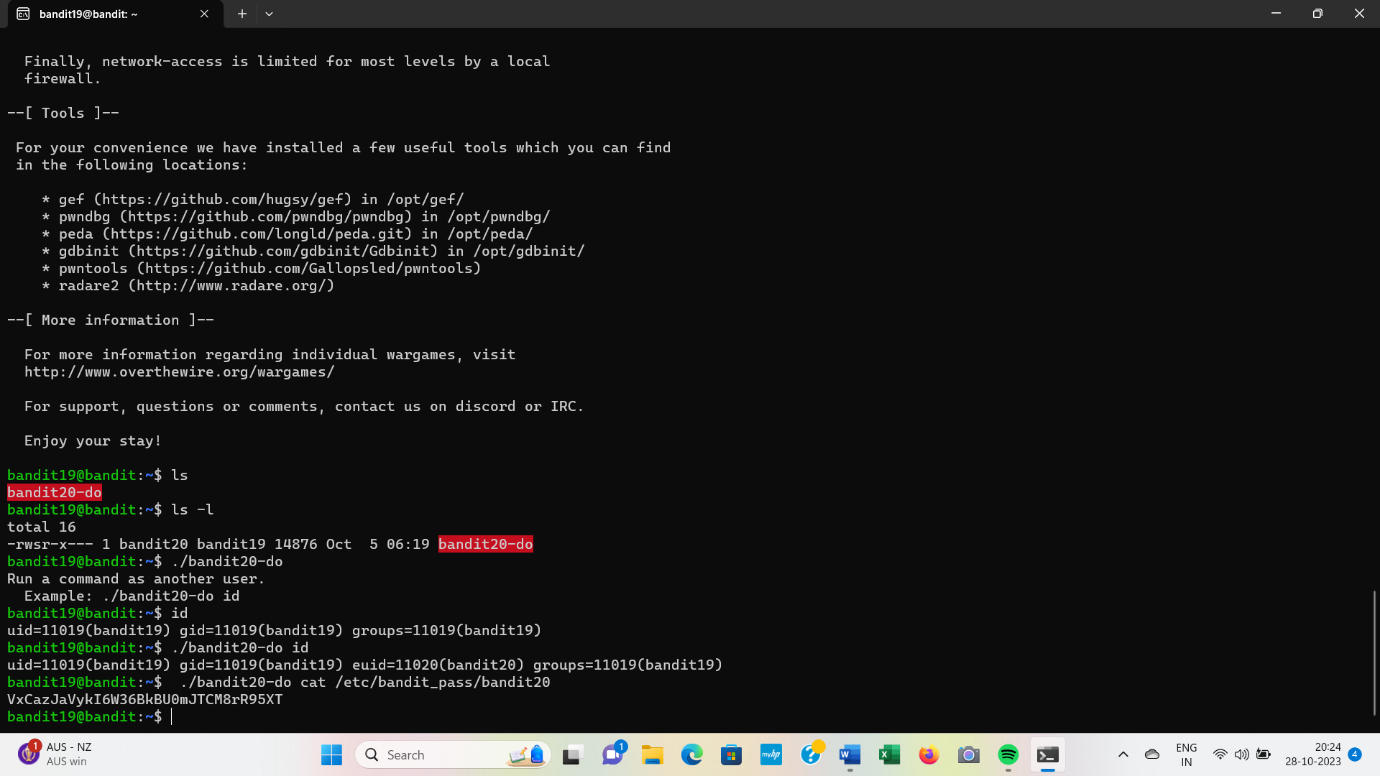
I ran "ls" I see that readme file the hint was talking about is available but we are not able to read it because I do not have permissions. I then moved on to see if I can find another way of accessing the file. I read the ssh manual to see if I can find a way to login without it kicking me out immediately. After reading the ssh manual I found 2 options , -T and -t. The other options were used to either tunnel, forward, or specify login attributes. I tried logging in with the "ssh [bandit18@bandit.labs.overthewire.org](mailto:bandit18@bandit.labs.overthewire.org) -p 2220 -t “ bin/ssh” command and I was able to login without getting kicked out. The part where you write the commands was blank, I tried to ran "ls" and I was able to see the "readme" file and I was able to read it and see that the password for bandit19



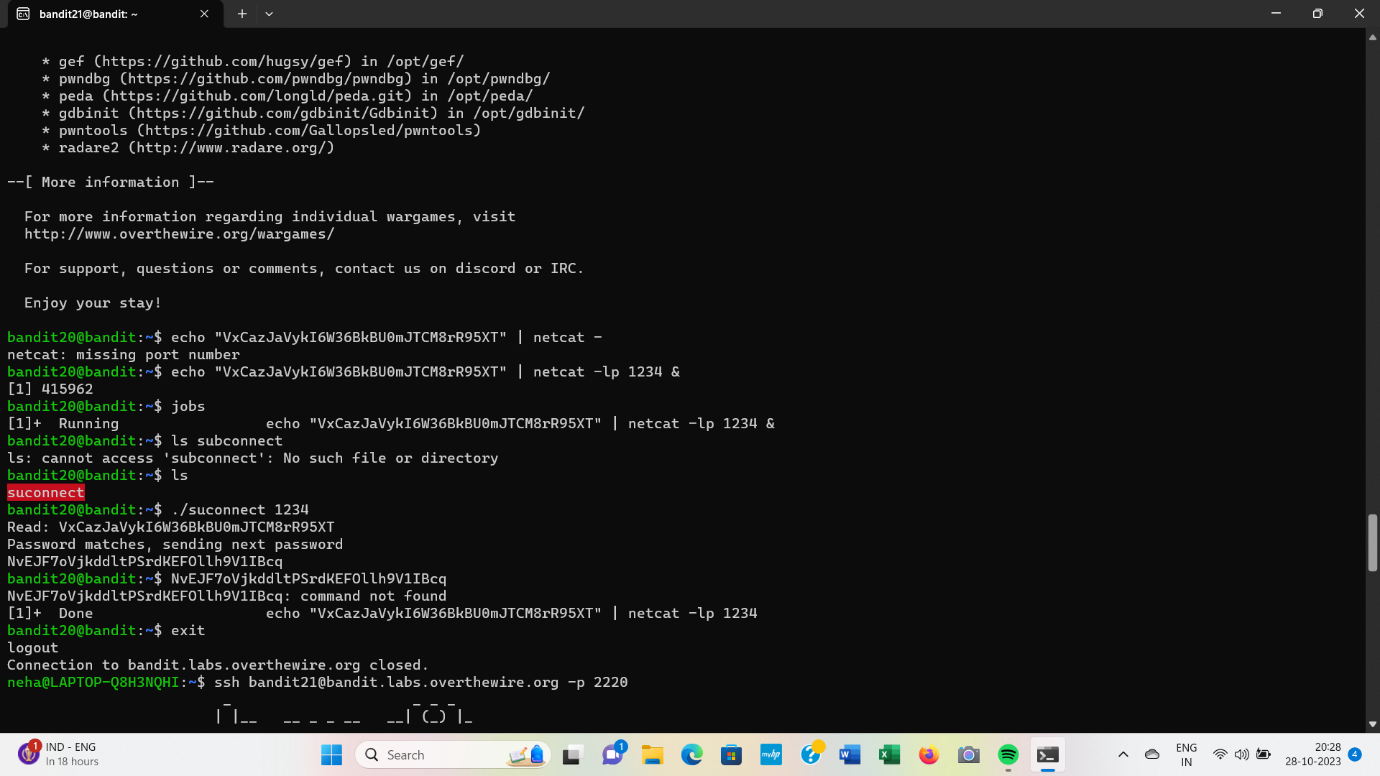
******

***Level 19***

After submitting password I received in level 18, I just ran ls command.I saw the binary file named “bandit20-do”. The hint stated that I needed to execute the file, so I did with the “./bandit20-do”command. The output said “Run a command as another user”. I then ran ./bandit20-do id and I was able to see bandit20s id information. Since the password for bandit20 is in the "/etc/bandit\_pass/" I decided to run “./bandit20-do cat /etc/bandit\_pass/bandit20” since only bandit20 is able to read the file. As a result ,I was that to see the next password

******

***Level 20***



***Level21***

To clear this level I used a series of commands ;among them one was cron which reads files in /etc/cron.d/directory .Rest of the commands were the ones which I have already mentioned earlier and even made use in previous levels. In order to use cron jobs, an admin needs to allow cron jobs to be added for users in the '/etc/cron.allow' file.-Sources from web(just for knowledge).

