Exploring Bandit Levels

De Silva K.R.K.D

Abstract

This report chronicles my playthrough of OverTheWire's Bandit stages, an entertaining examination of Linux security. From fundamental commands for Linux to complex security and cryptography problems, I provide knowledge, tactics, and solutions at every stage. This report shows my development and flexibility as I move through the Bandit levels in the cybersecurity industry. For those exploring novel tasks or ready to set off on their own adventure, it is an invaluable resource I cordially invite you to come with me on this fascinating journey, where each level advances our progress toward becoming adept keepers of digital forts. This abstraction offers a brief overview of the most important findings and training made along the advancement of Bandit levels, from first discovery to expert impact strategies.

Table of Contents

Introduction to the topic

A necessity in the constantly changing field of cybersecurity is being able to secure Linux computers. In this field, practical knowledge, strong problem-solving skills, and a thorough grasp of system vulnerabilities and exploits are essential. Essential. Enter the Bandit levels, a gripping set of tasks carefully created by overtheWire to engross both novices and veterans in the area of Linux security. Bandit levels are fundamentally a place of play for hacking ethics, a digital testing field where competitors are exposed to a variety of Linux security situations, from the fundamentals of shell scripting and system administration to the complexities of increasing privileges and cryptographic enigmas.

My quest begins with a modest introduction to the fundamentals, unraveling the complexities of system permissions and interpreting basic Linux commands. Each level offers a different difficulty as I advance, like a puzzle that needs to be solved. Whether it's taking advantage of configuration errors, getting around access limitations, or cracking cryptographic ciphers, I explain my solutions and offer a thorough manual for individuals who want to overcome these obstacles on their own.

However, this study extends beyond simple fixes; it captures the heart of training in cybersecurity and the practice of ethical hacking. It emphasizes the value of accurate records and the continuous search of information in the quest for mastery. I hope to inspire and illuminate by the sharing of my experiences, demonstrating that anyone is able to begin on a similar road of learning and development throughout the cybersecurity field if they have the will and the necessary tools.

However, this study aims to go beyond only fixing issues. It captures the very core of ethical hacking and cybersecurity pedagogy- an attitude that emphasizes the practice of thorough documentation, celebrates the musical harmony produced by teamwork, and reverses the never-ending pursuit of knowledge that acts as the oven of expertise. Our goal in sharing our personal stories on these pages is not just to chronicle our experiences, but also to motivate and instruct.

I encourage you, my valued reader, to go with me as I navigate the convoluted passageways of Bandit, removing the veils of mystery that envelop Linux security layer by layer. Whether you are an adventurous beginner, an experienced guardian looking for fresh challenges, or simply an interested bystander, I can guarantee you that this report will not only offer insightful insights but will also heighten your understanding of the complex web ethical hacking. I go closer to becoming alert guards of digital fortresses with each challenge I successfully complete. Consequently, I cordially invite you to read my riveting account, "Exploring Bandit Levels."

Methodology

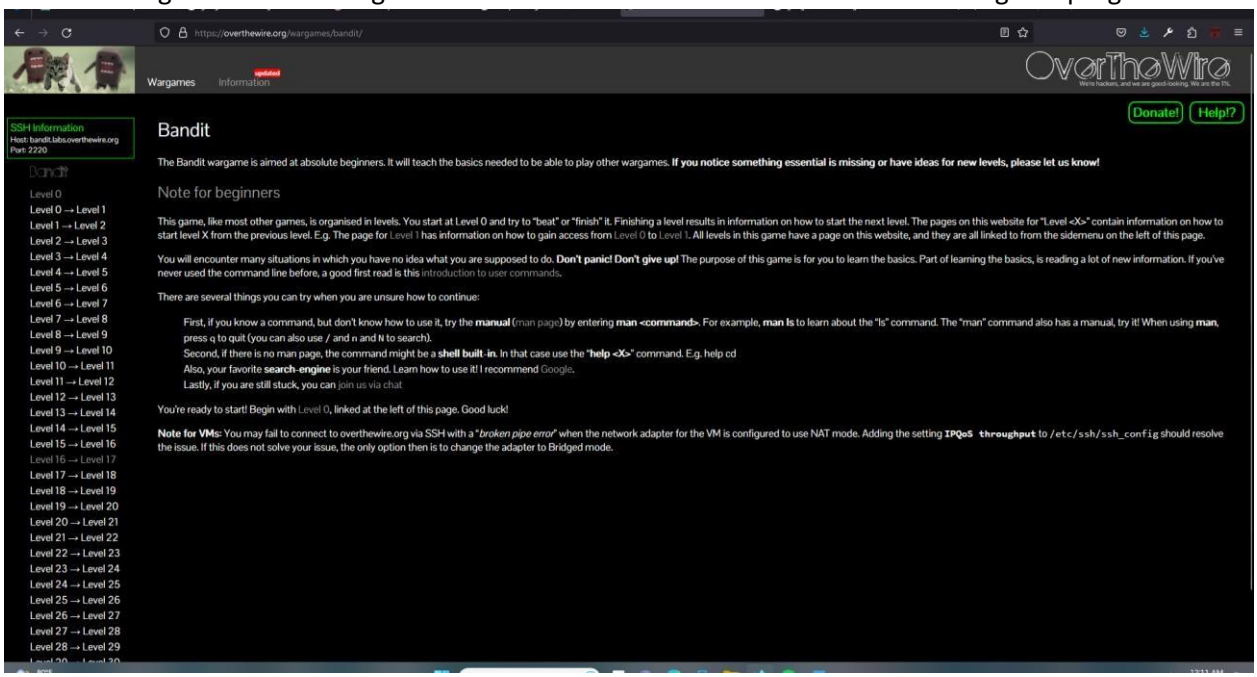
We take a thorough and organized approach to navigating the Bandit levels on OverTheWire. We start by carefully examining the guidelines and requirements for the first level to make sure we understand the goals at hand. Setting up a separate, safe Linux environment for testing is essential since it enables us to explore freely while preserving system integrity. At the same time, I keep comprehensive records of my progress, revelations, and responses to compile a thorough logbook of my adventure. As I examine every stage, I use critical thinking to uncover the challenge's core, spot possible vulnerabilities, and learn about the system's complexity. I put a lot of focus on lifelong learning, doing research when faced with new ideas, and using discussions, instructions, and internet tools to further my expertise. During this method, automation and scripting proved to be vital allies in cutting routine tasks and increasing my productivity.

I take on every problem with determination and perseverance, knowing that mistakes can teach us important lessons. Each level is finished at the end of my tour, and I provide thorough documentation of my answers and lessons learned. I continually evaluate my development, realizing how each level builds on the information learned from earlier ones, and I use this knowledge to take on the increasingly difficult challenge. The last step entails putting my solutions, observations, and progress diary into a thorough report. This report serves as both a personal account of my experiences and a resource for the larger community, facilitating information sharing and advancing our understanding of Linux security. In the end, my approach guarantees a methodical and instructive study of the Bandit levels, encouraging skill advancement and individual development in the field of ethical hacking and cyber security.

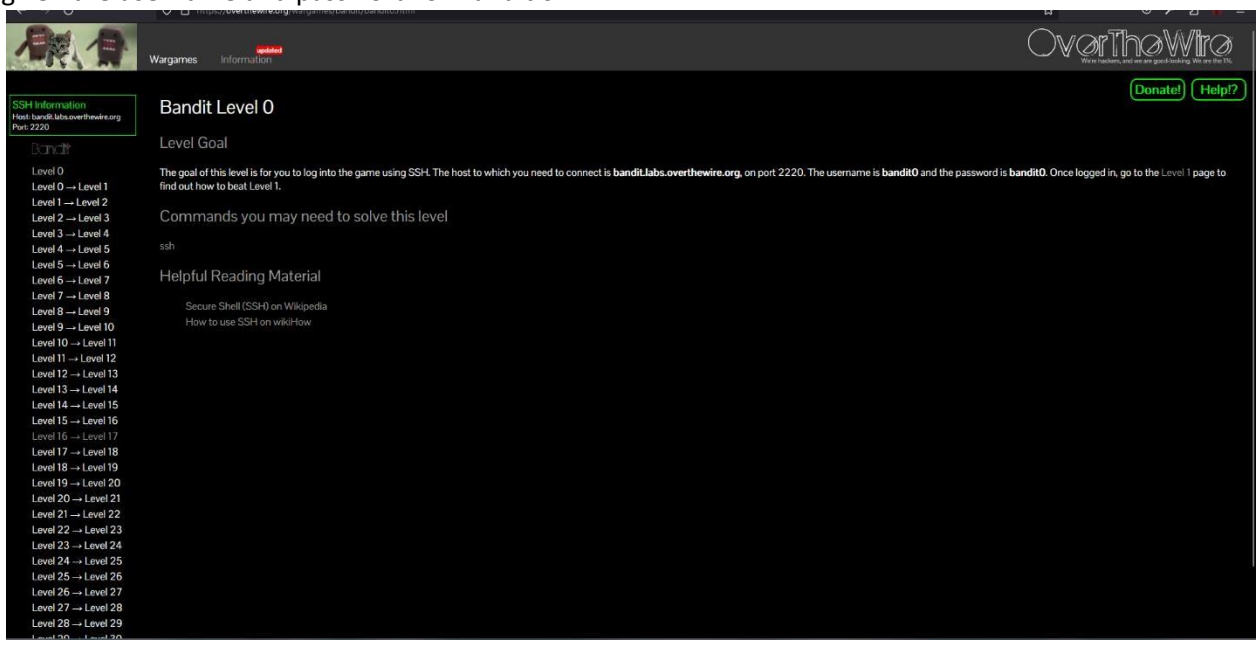
Bandit levels and solutions

Bandit0

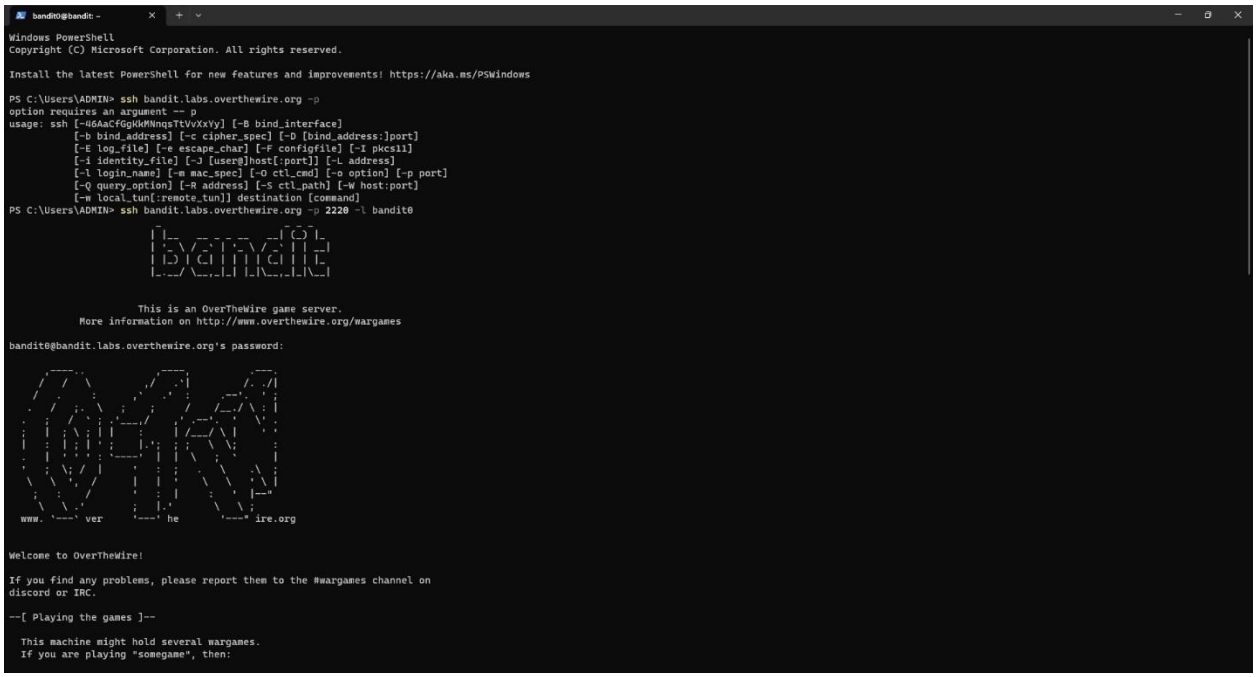
Before starting Bandit 0 we are given a brief introduction about Bandit and how the game progresses.



Here we can continue the Bandit game on Windows or Linux. Must log in via SSH to start. We have been given the username and password for Bandit 0.



Type, "ssh bandit.labs.overthewire.org -p 2220 -l bandit0" and use the given password and log Bandit0.

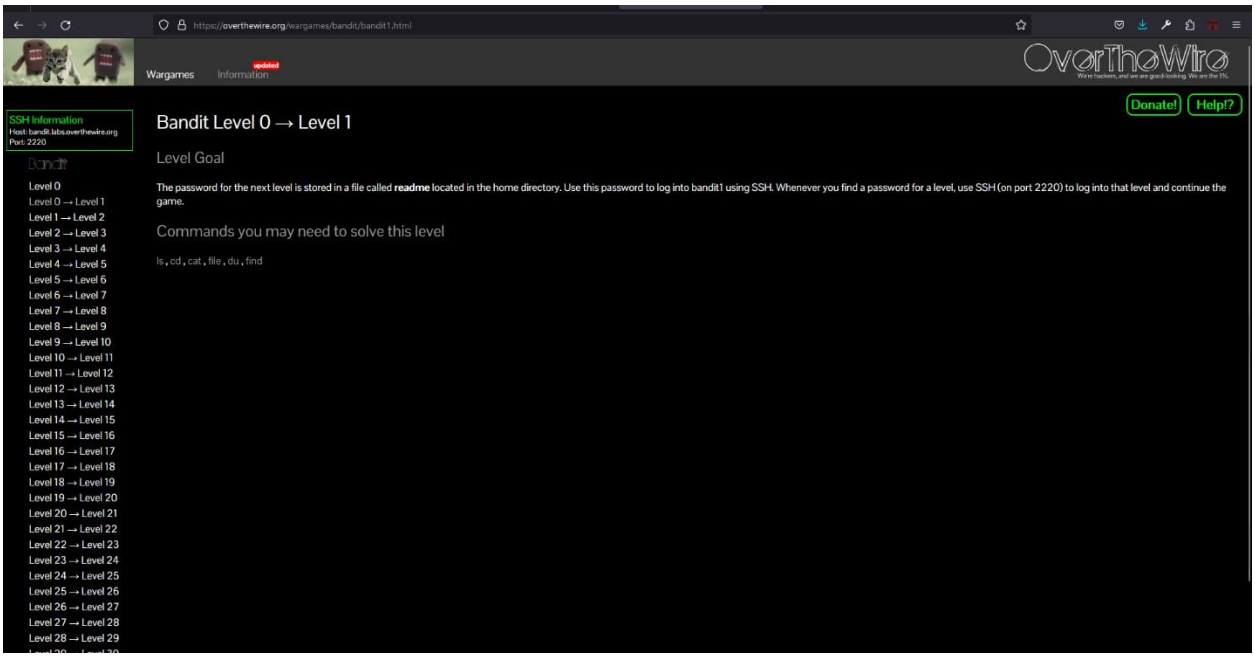


Bandi0 -> Bandit1

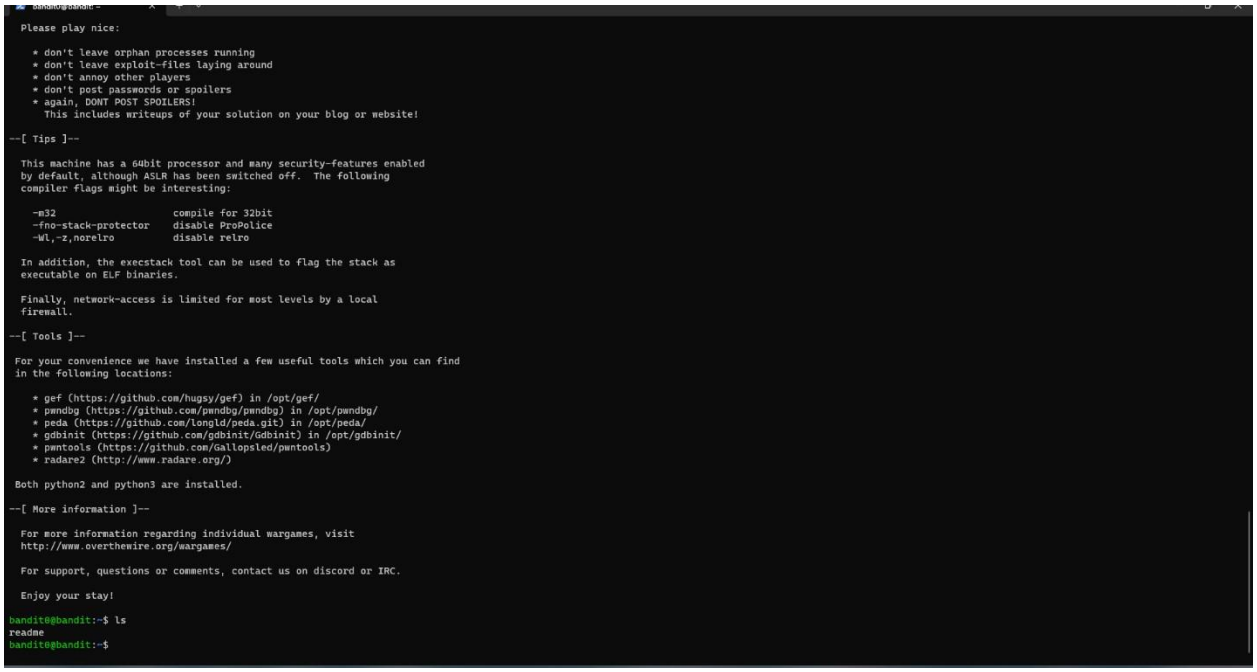
Once we log Bandit0 we need to find the password of Bandit 1. They give us a hint and some commands.

Here are those commands.

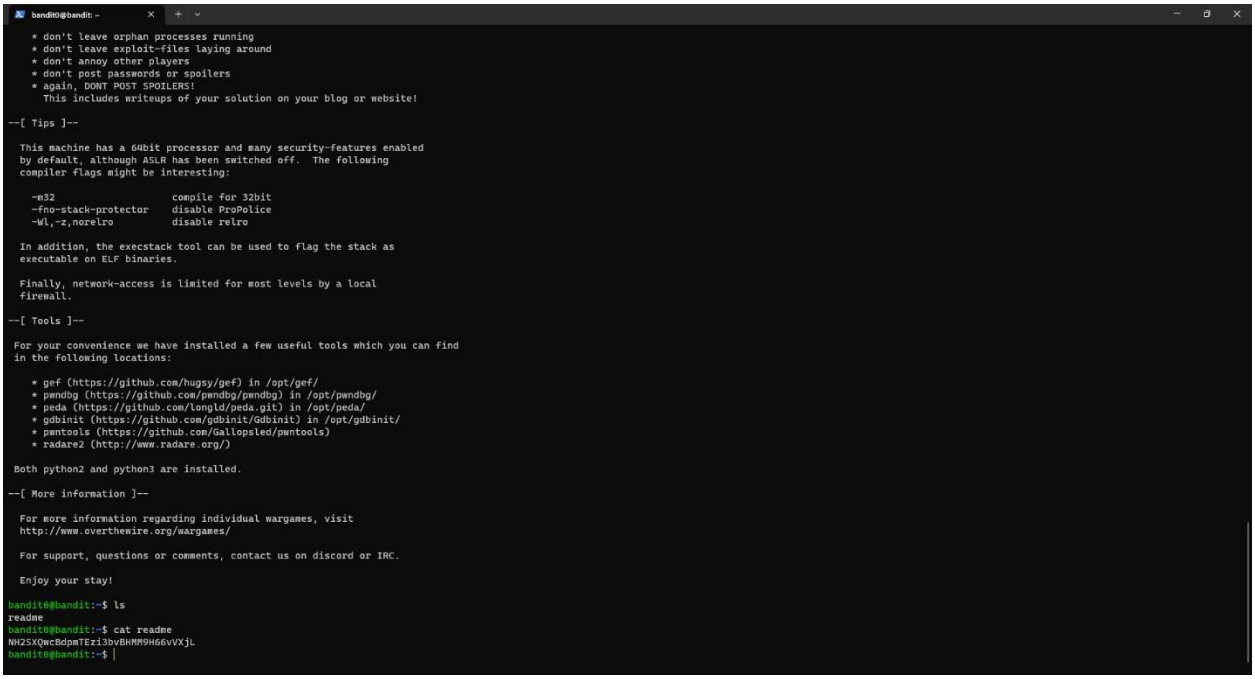
- ls – list directory
- cd – change the working directory
- cat - print the content of a file onto the standard output stream.
- file – determine file type
- du – measure the disk space occupied by files or directories.
- find - search for files in a directory hierarchy.



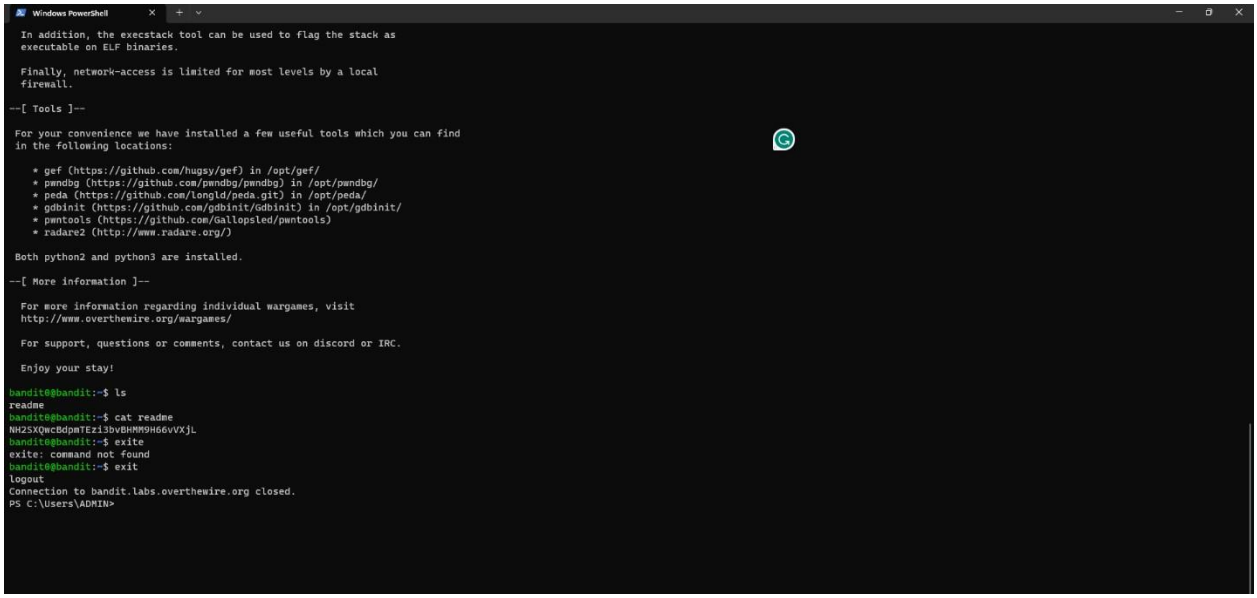
Use the “ls” command to see the file readme.



Run “cat readme” to see the contents of the readme and to get the password.



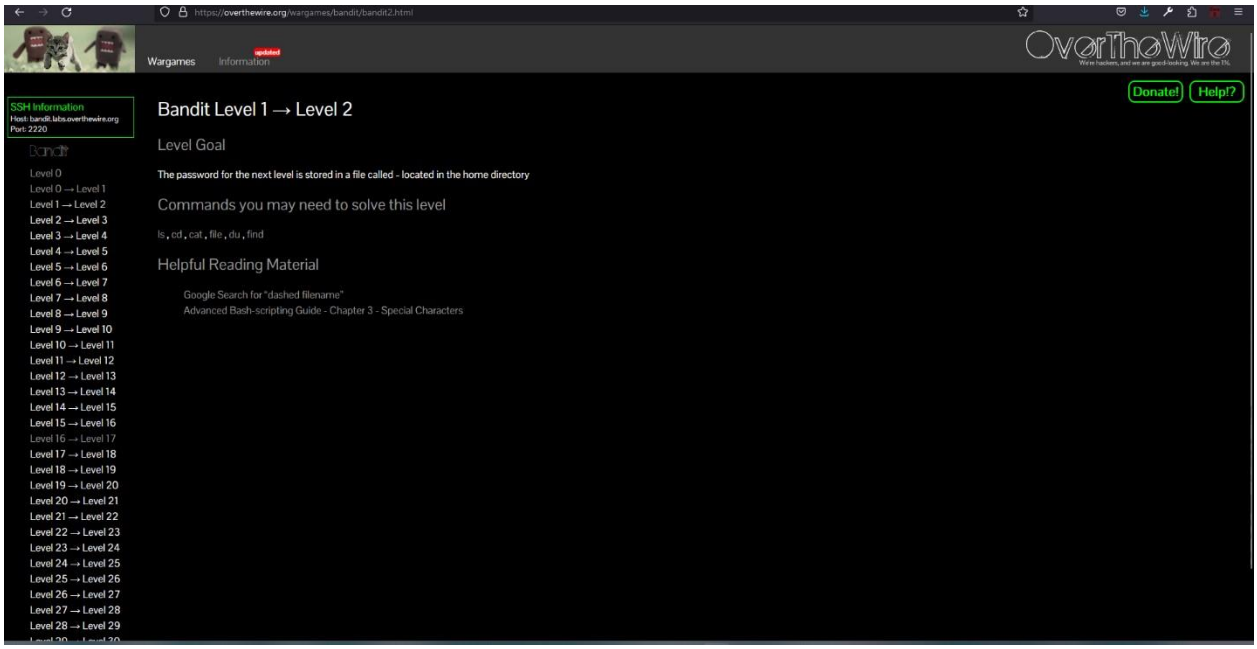
To logout, run “exit”



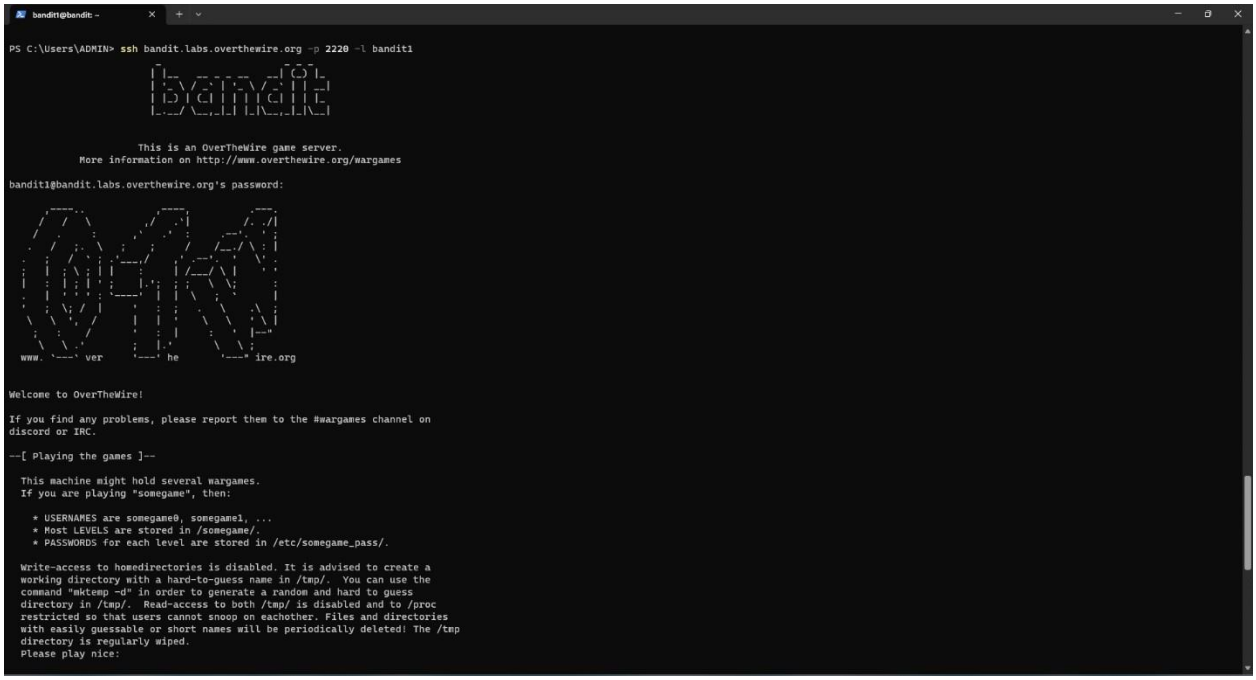
-

Bandit1 > Bandit2

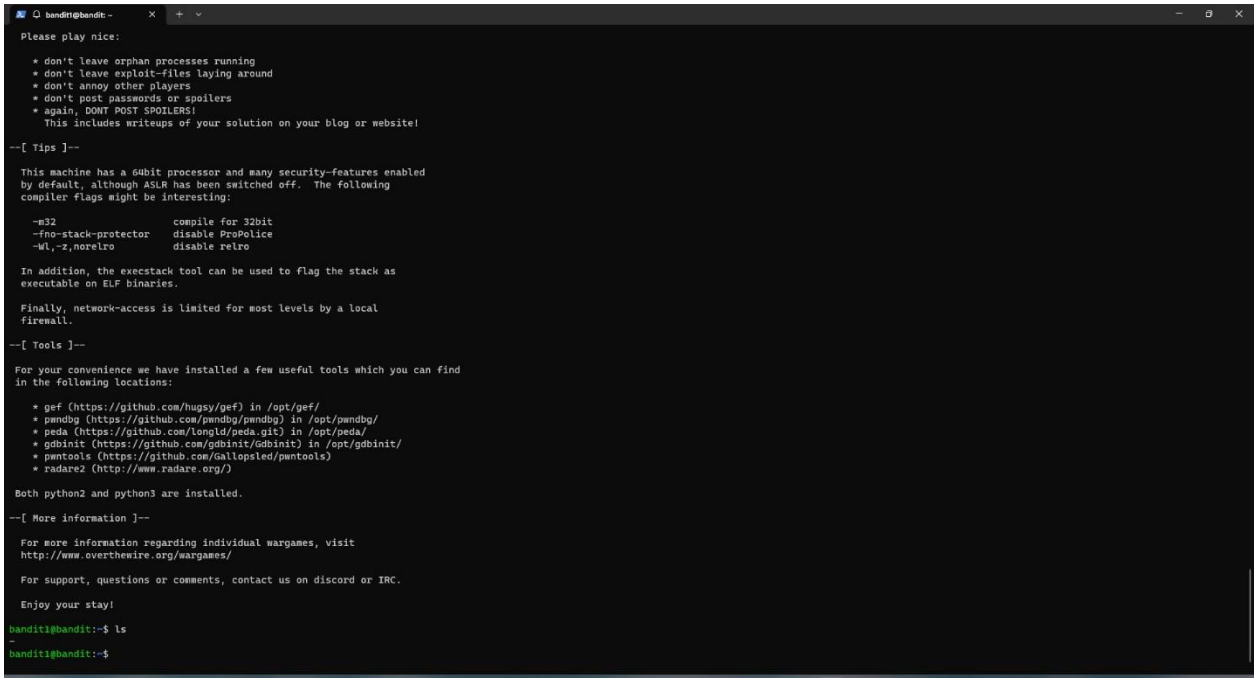
Read the hint and try to get an idea.



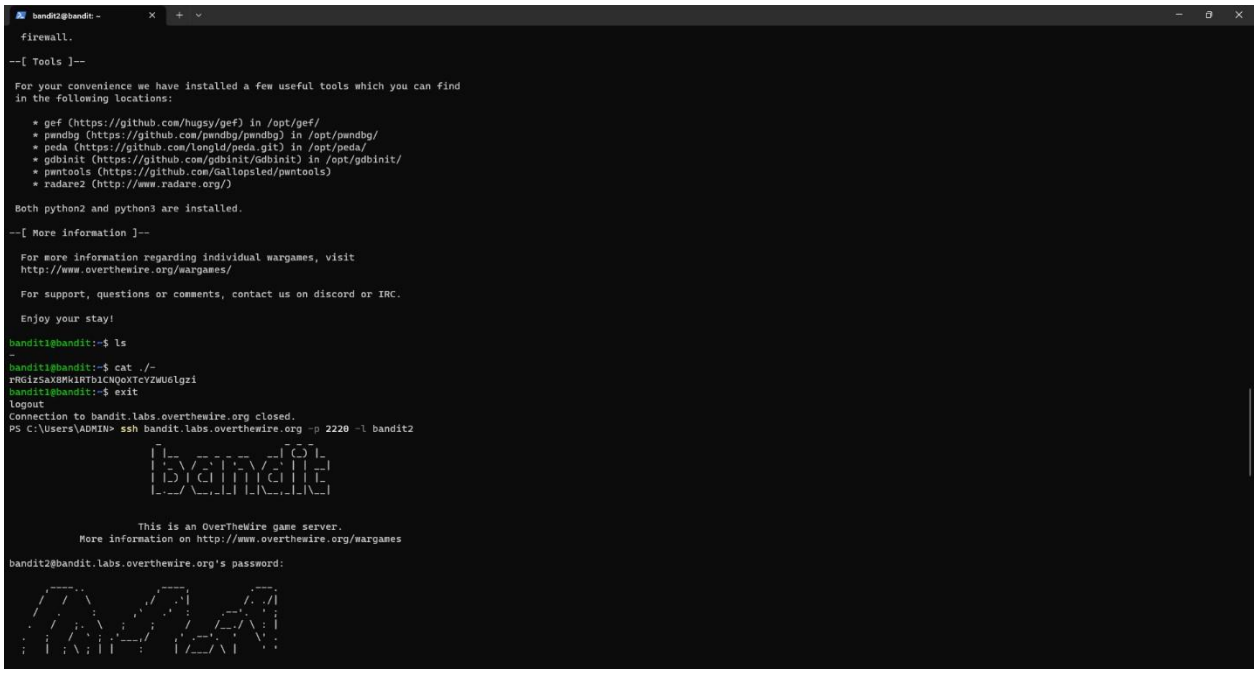
Now log in to Bandit1, from the found password.



Run the “ls” command and find the “-“ file



Use the “cat” command to find the Bandit2 password. But we cannot use the cat command and only “-“ because the system thinks “-“ is a command. So we need to use the cat command within “./-“ these commands and get the password.

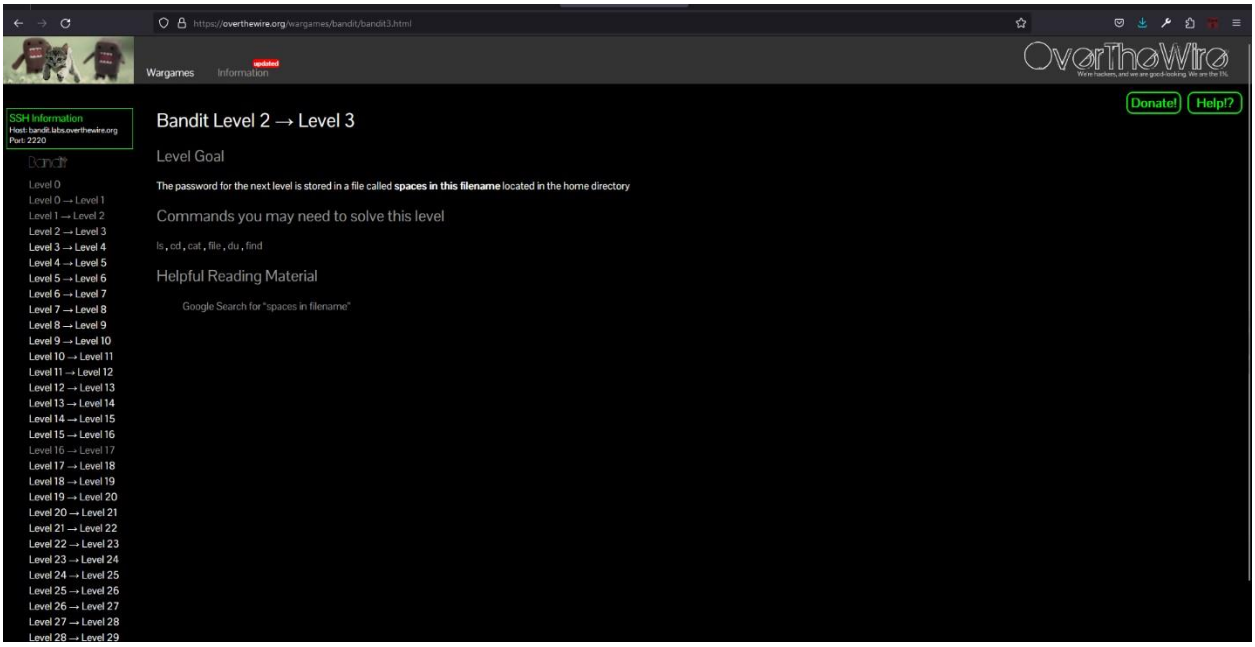


Log out using the “exit” command.

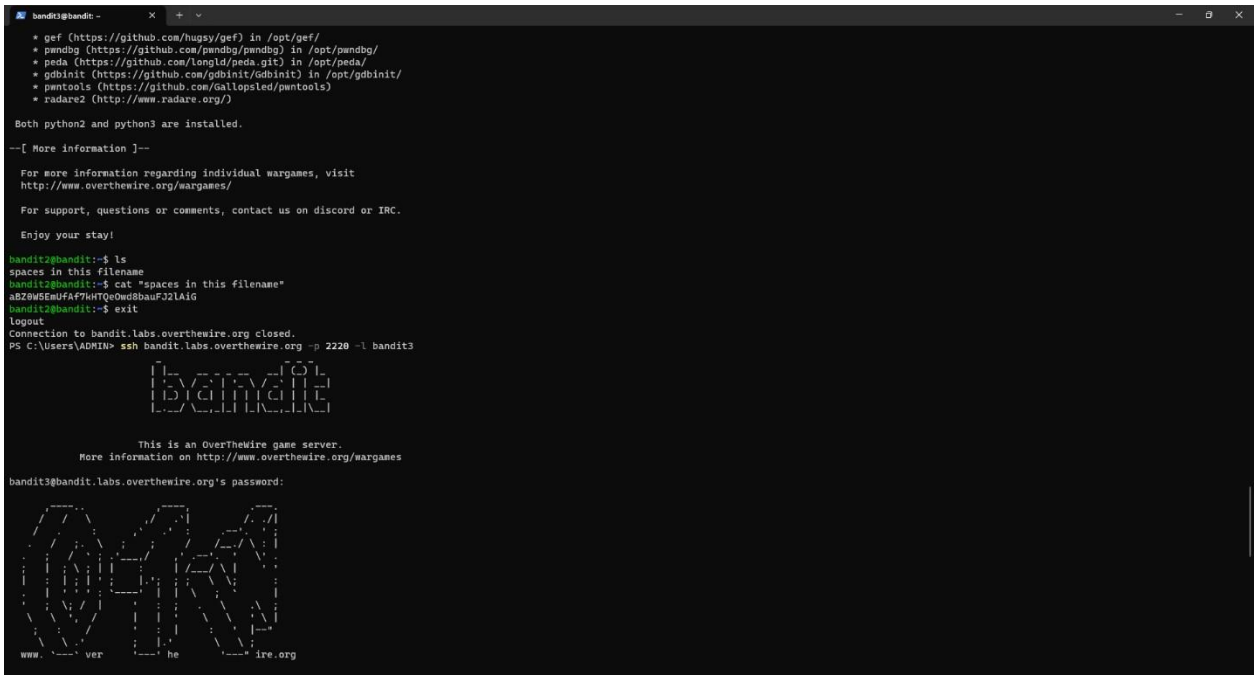
Bandit2 > Bandit3

-

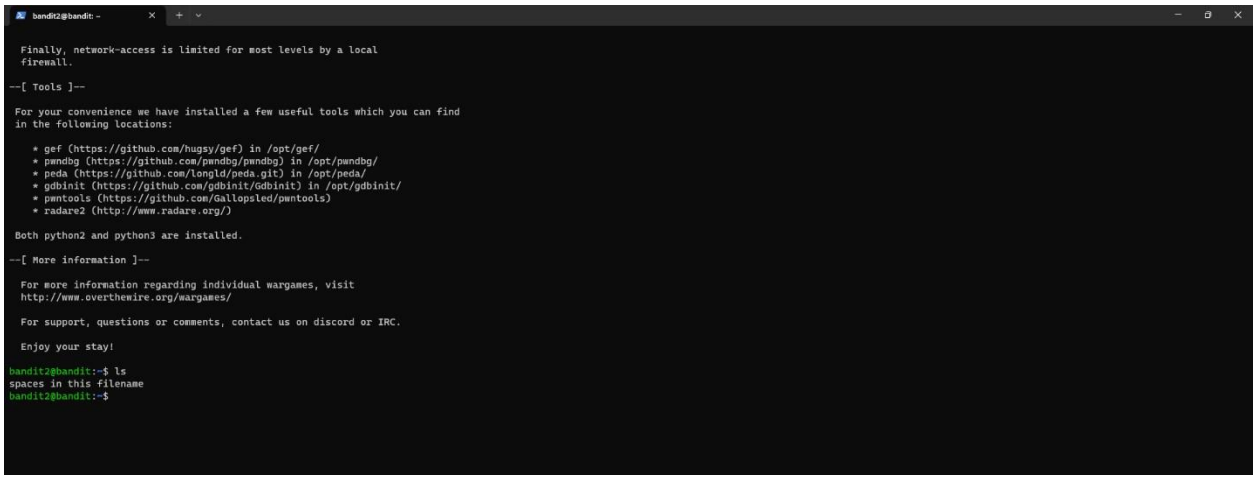
They tell us the next password is in a “spaces in this filename” file.



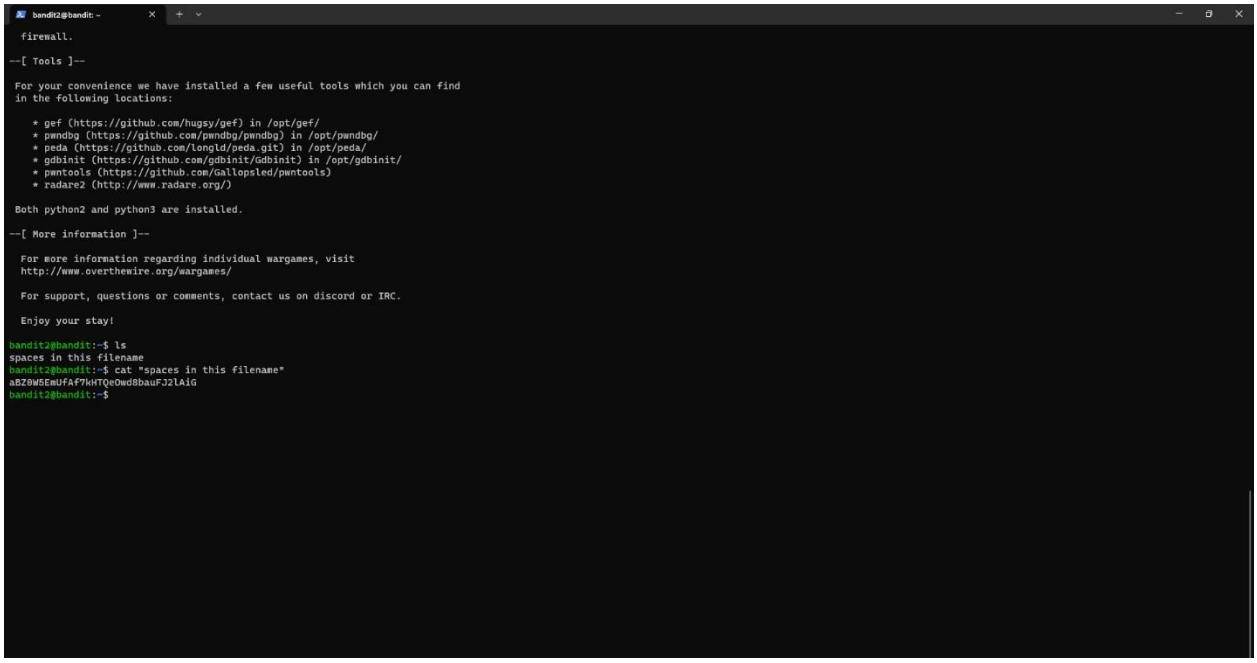
Log into Bandit2 using the username and the password.



Using the “ls” command find the file name.



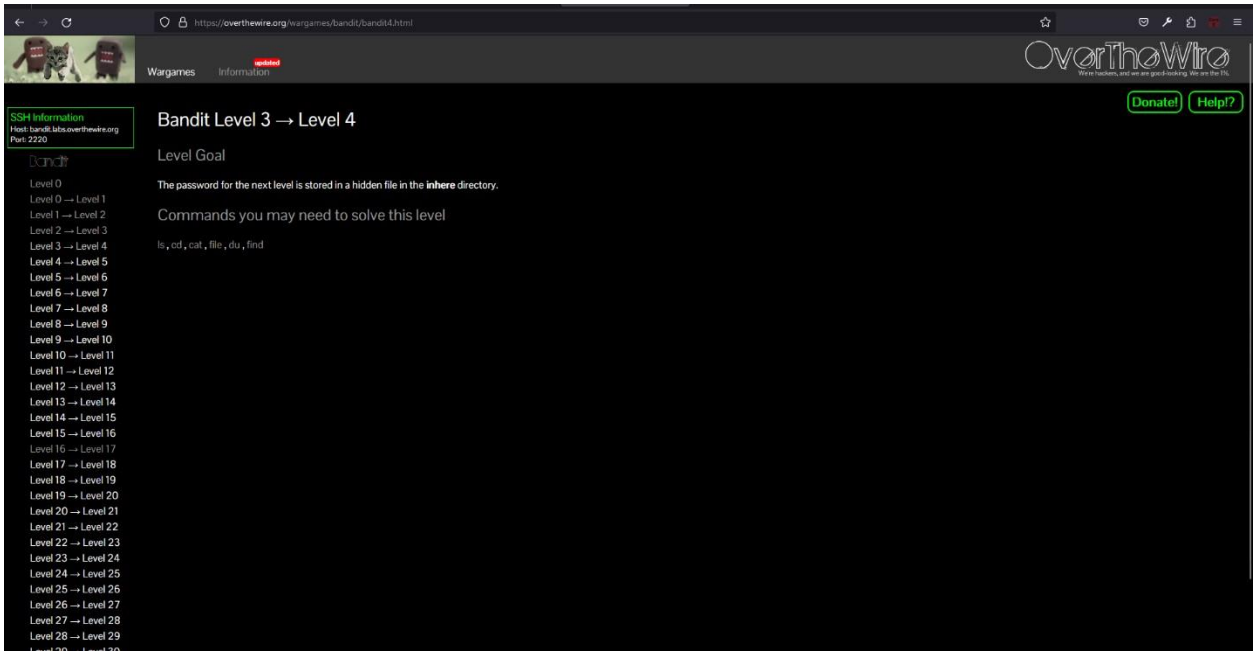
Run the “cat” command with the file name. You can get the Bandit3 password.

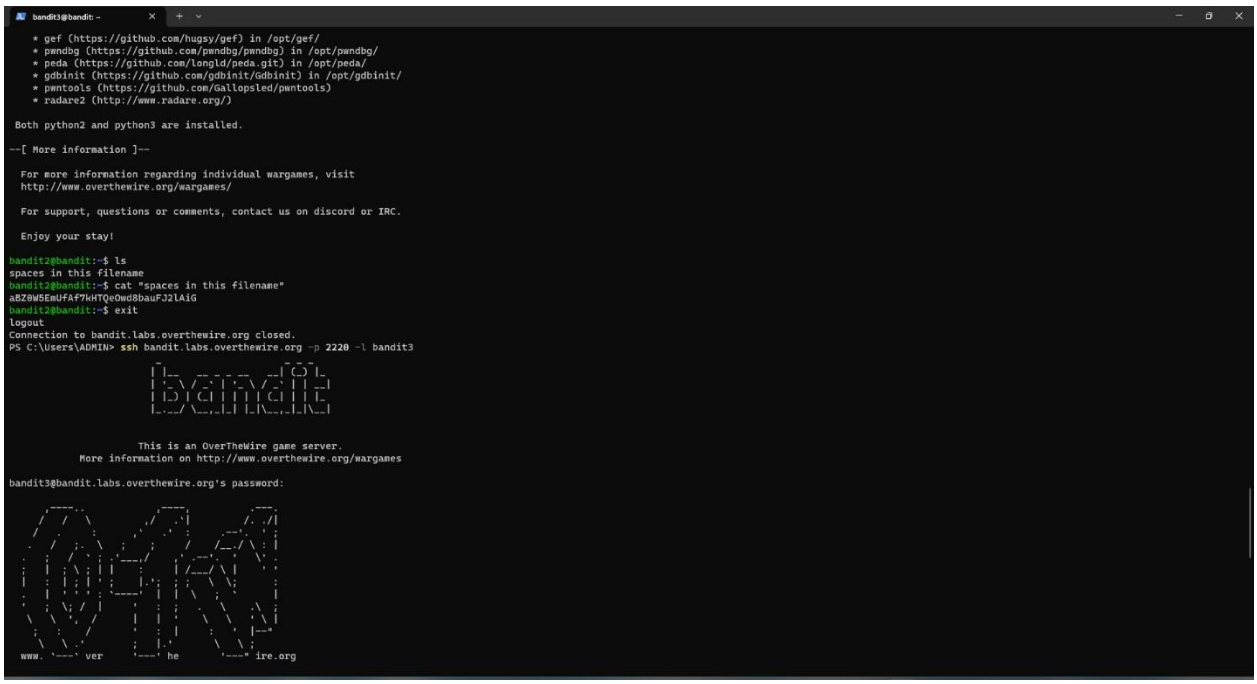


Bandit3 > Bandit4

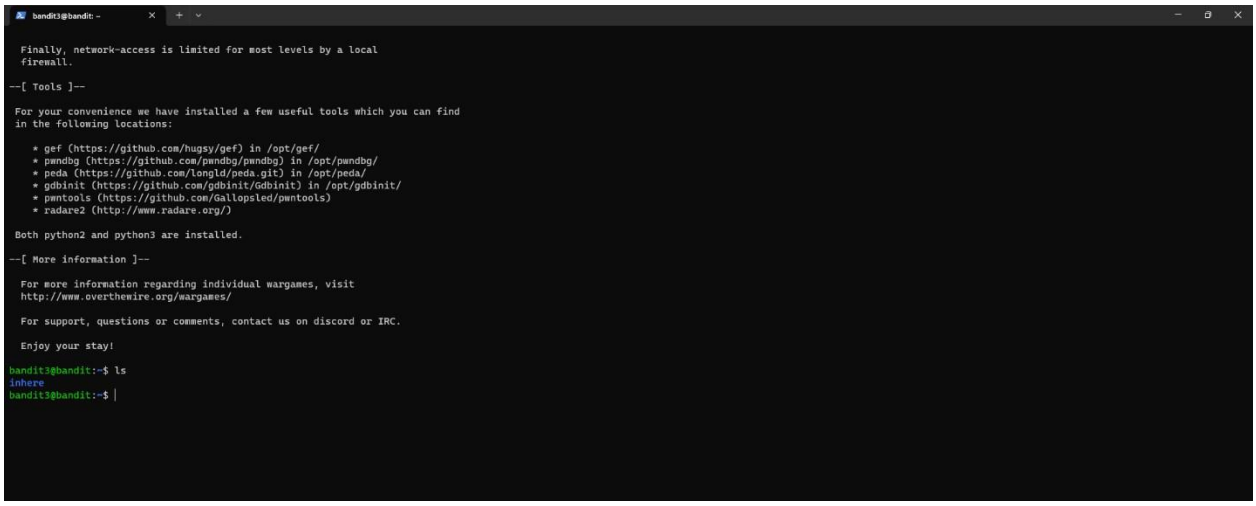
Log Bandit 3 to use the password.

-

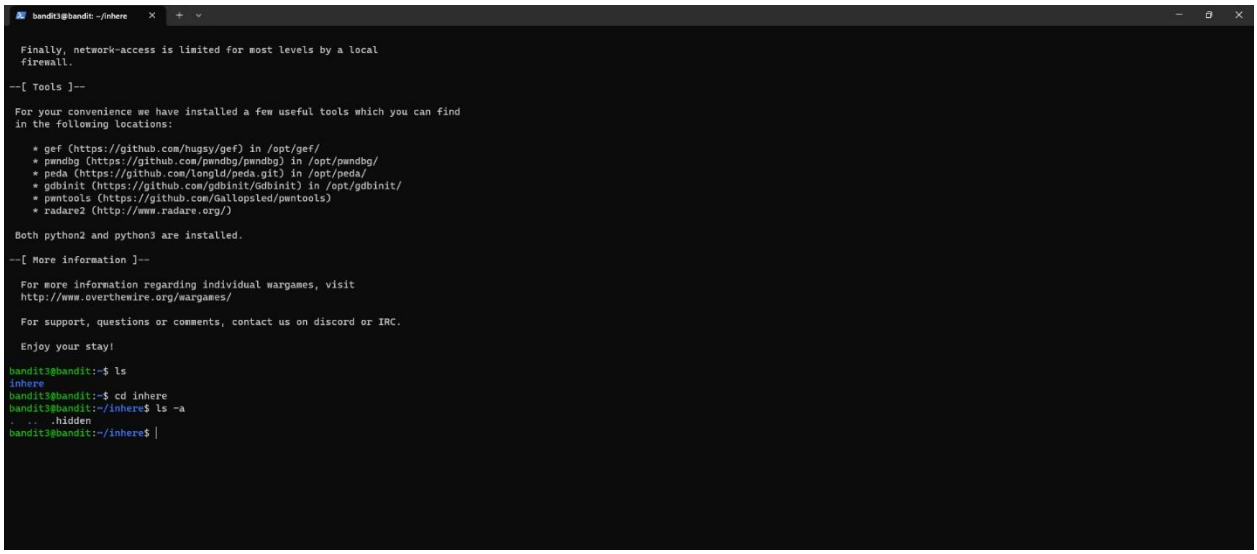




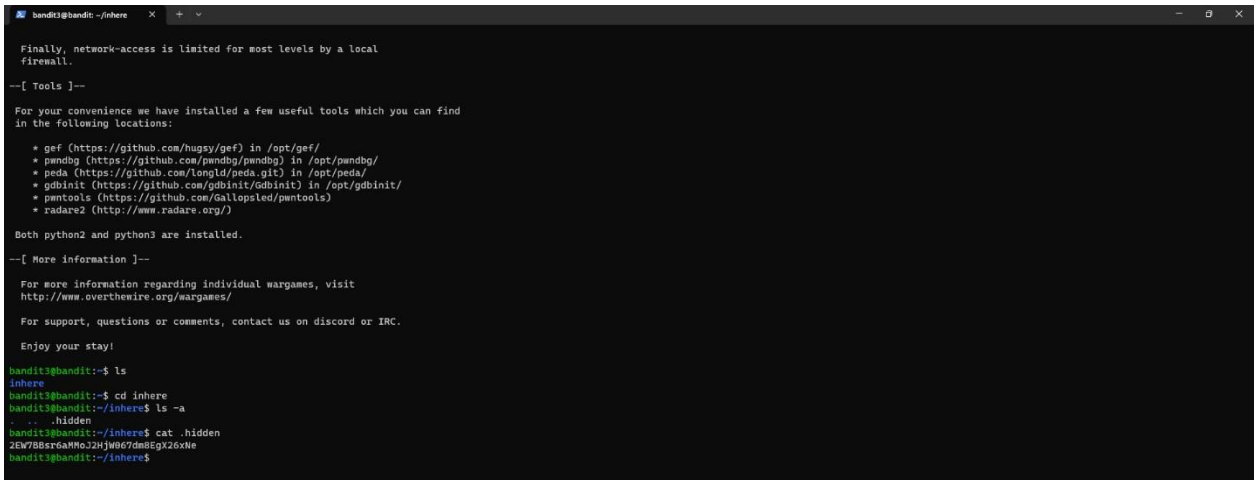
The next level password is hidden in the inhere file. Using the “ls” command find the “inhere” file name.



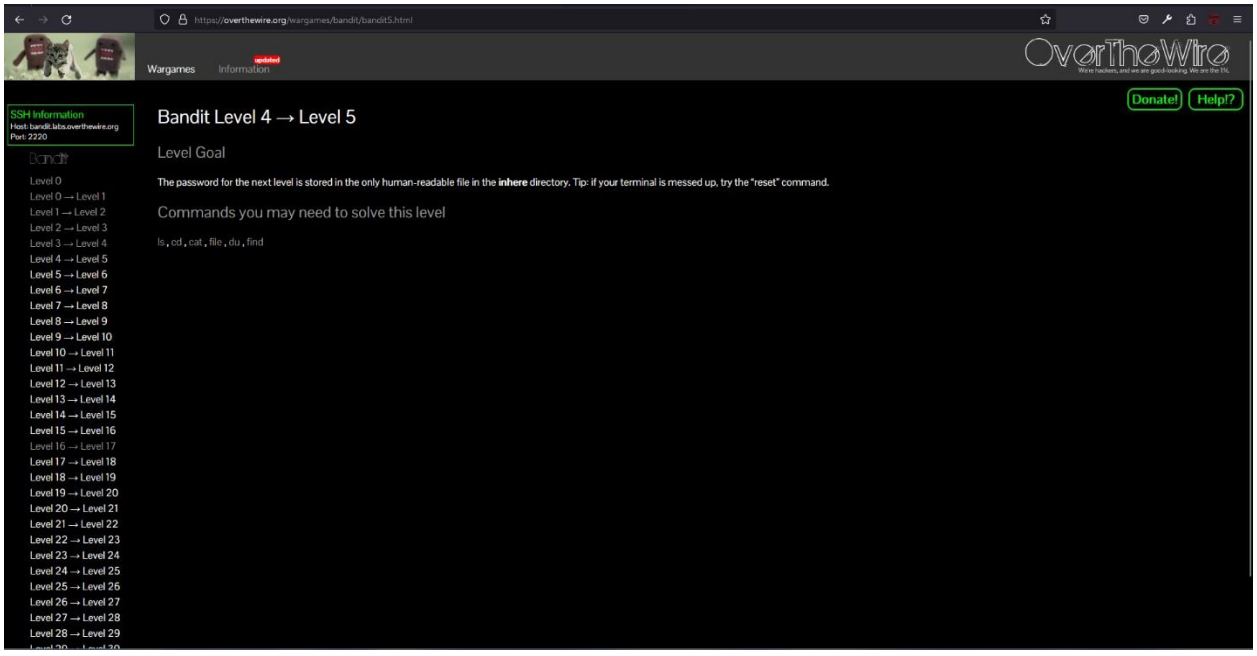
Using the “cd” command change the directory. Only non-hidden files are displayed by the “ls” command. With the “-a” flag, however, it displays all files, including hidden files.



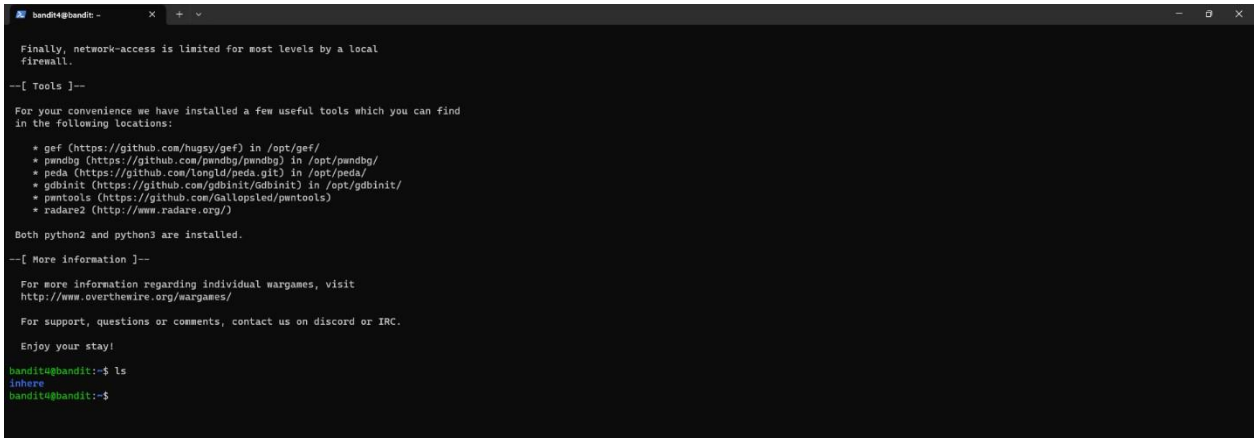
We can read the contents of the file because it is named “.hidden” and includes the password.



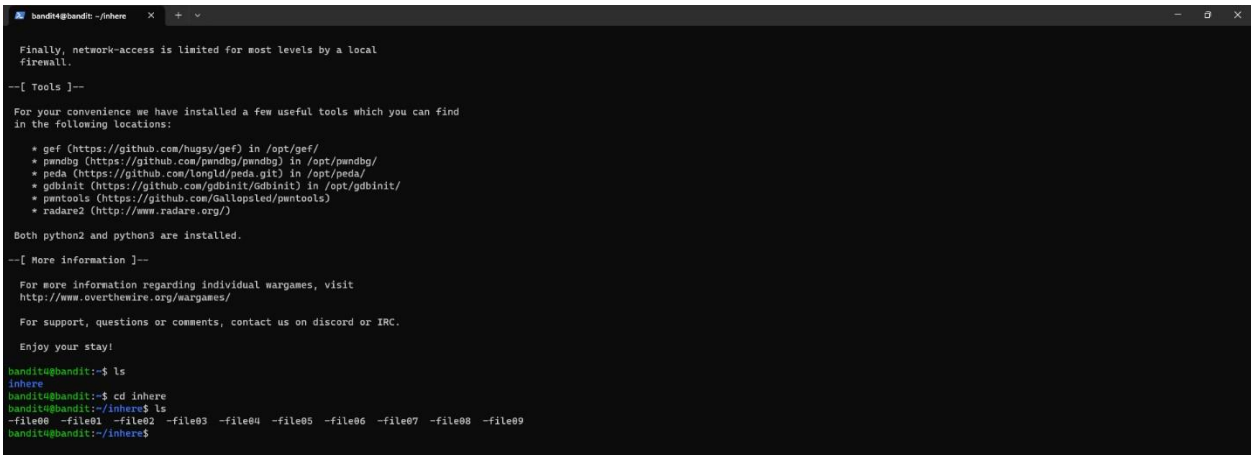
Bandit4 -> Bandit5



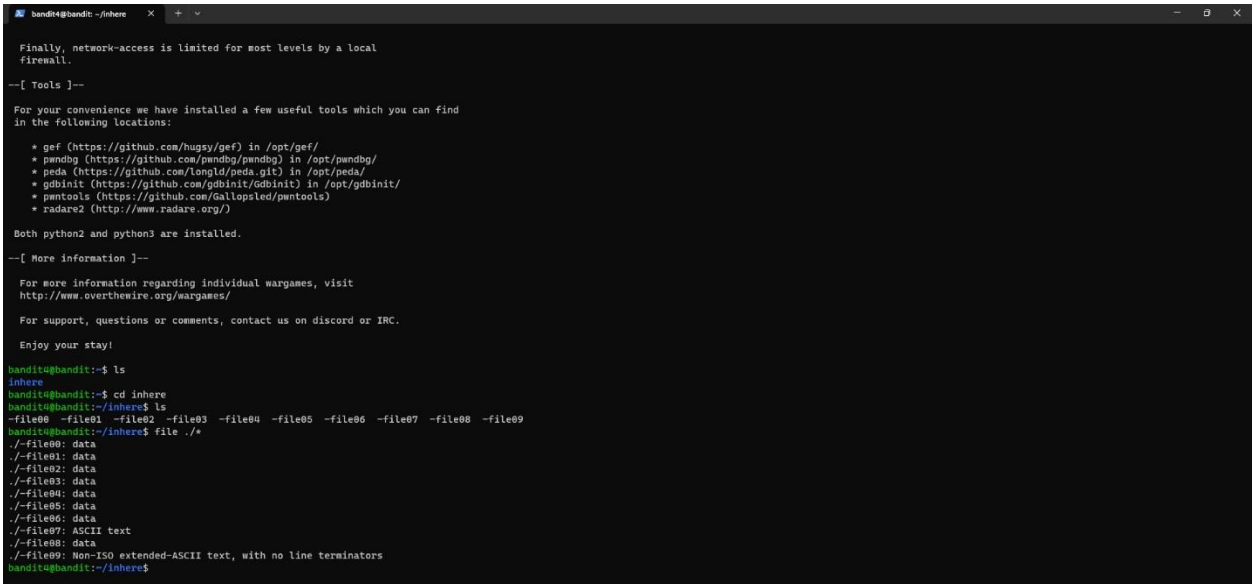
Log into the Bandit4 using the password. Use the “ls” command and find the file.



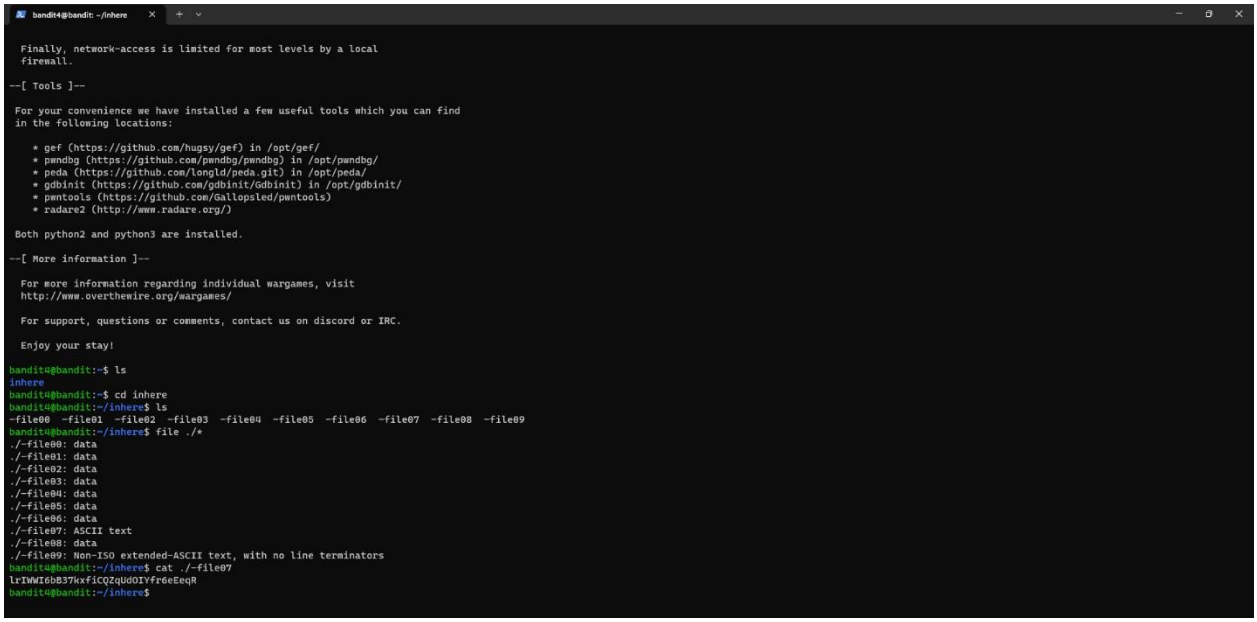
Using the “cd” command change the directory. And use the “ls” command.



Type “./-file*” to get a list of all the files in the directory along with their data types.

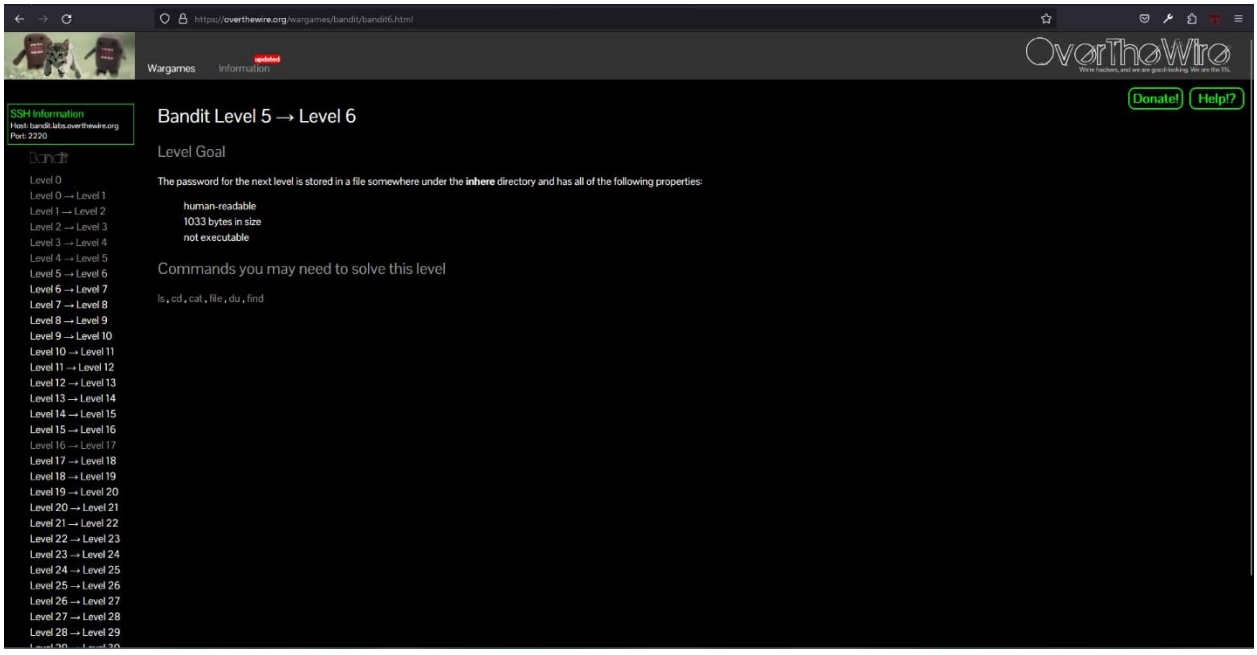


The “-file07” has ASCII text. Type “cat ./-file07” to get the password of Bandit5.

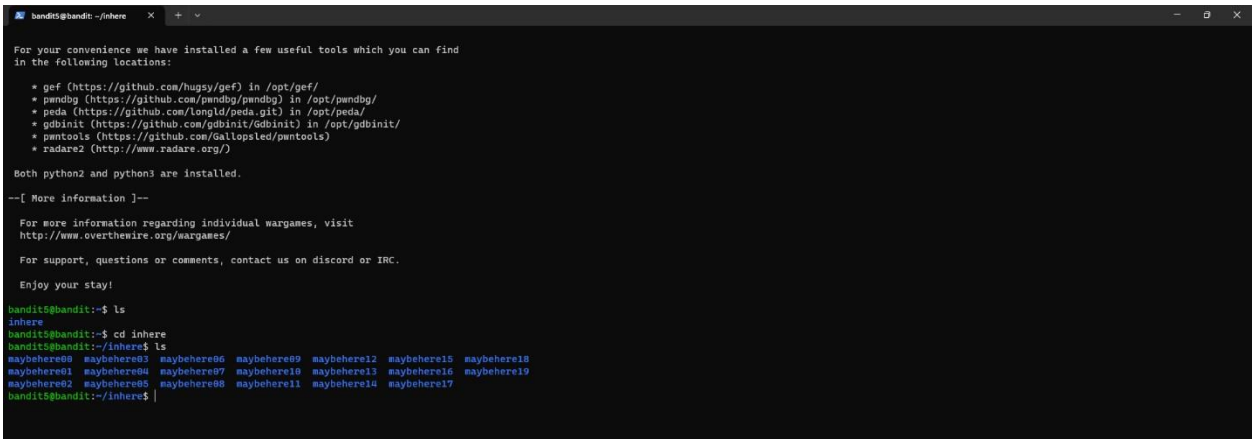


Bandit05 -> Bandit06

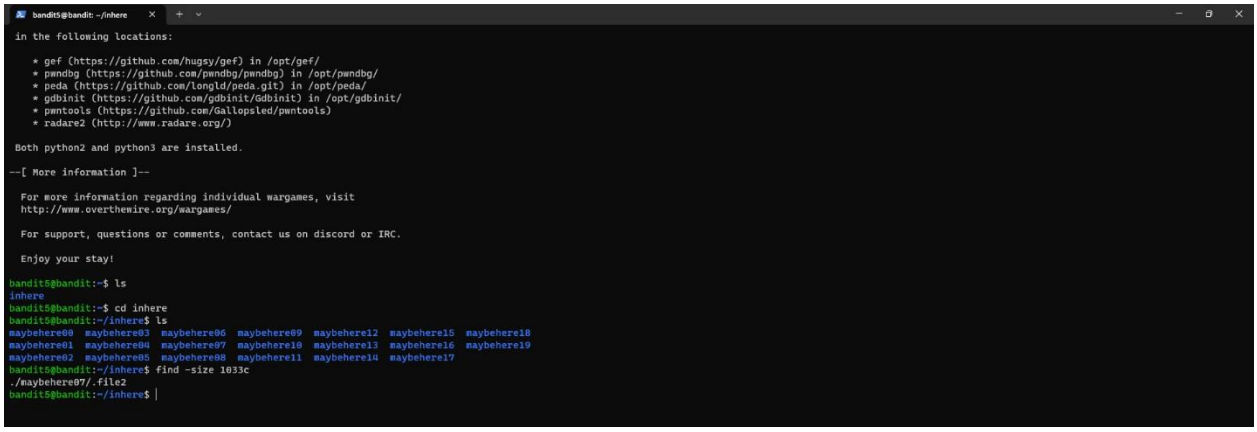
Log in to Bandit05 using the password.



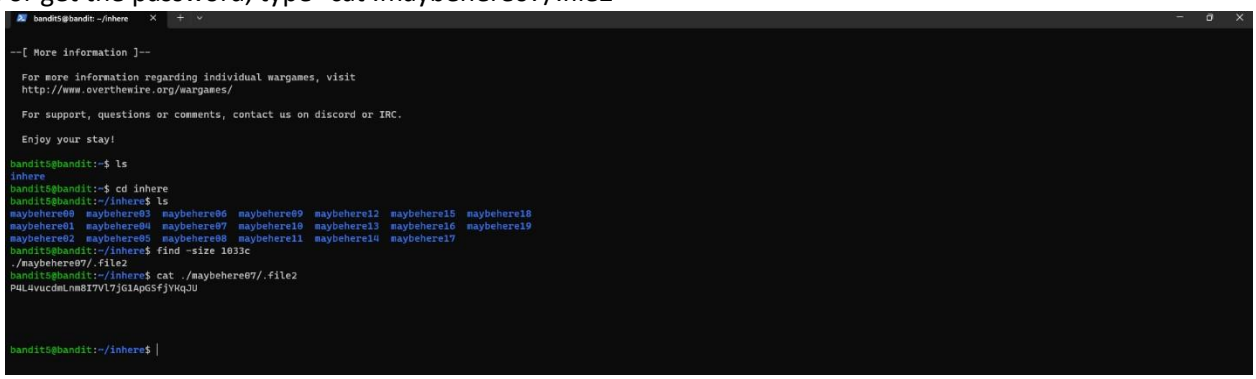
Type the “ls” to find the file name. Next type the “cd inhere” to change the directory. Again type the “ls” to list directory.



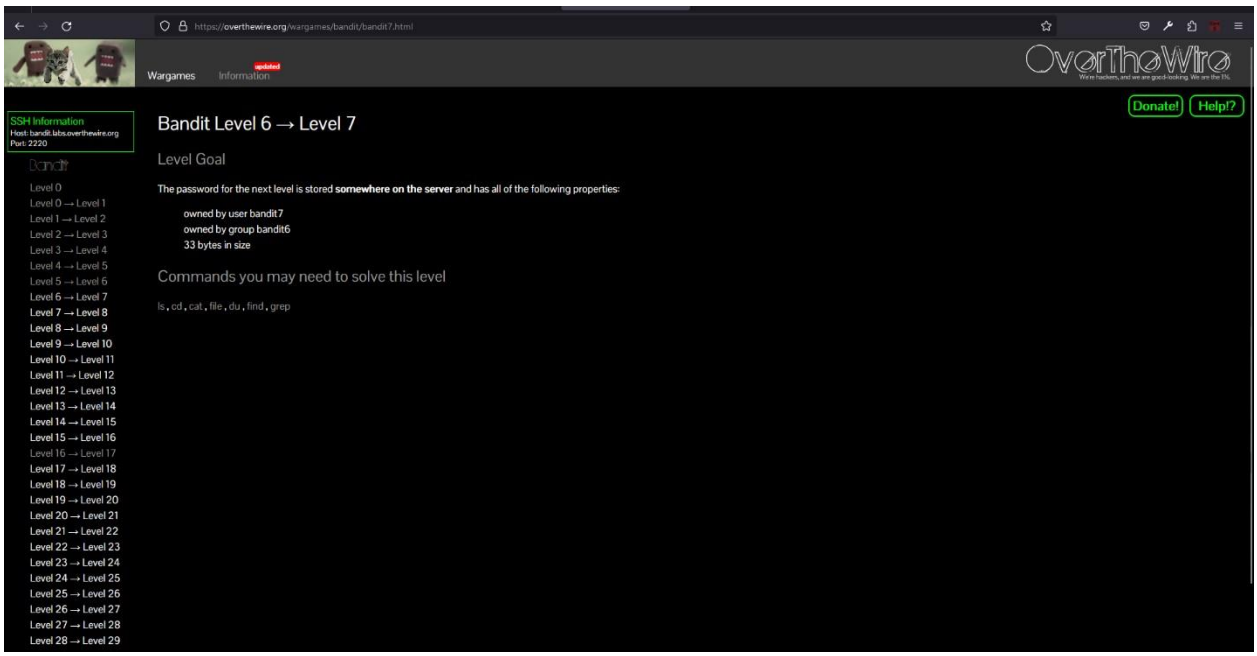
Type the “find -size 1033c” to find files that are readable with a size of 1033c.



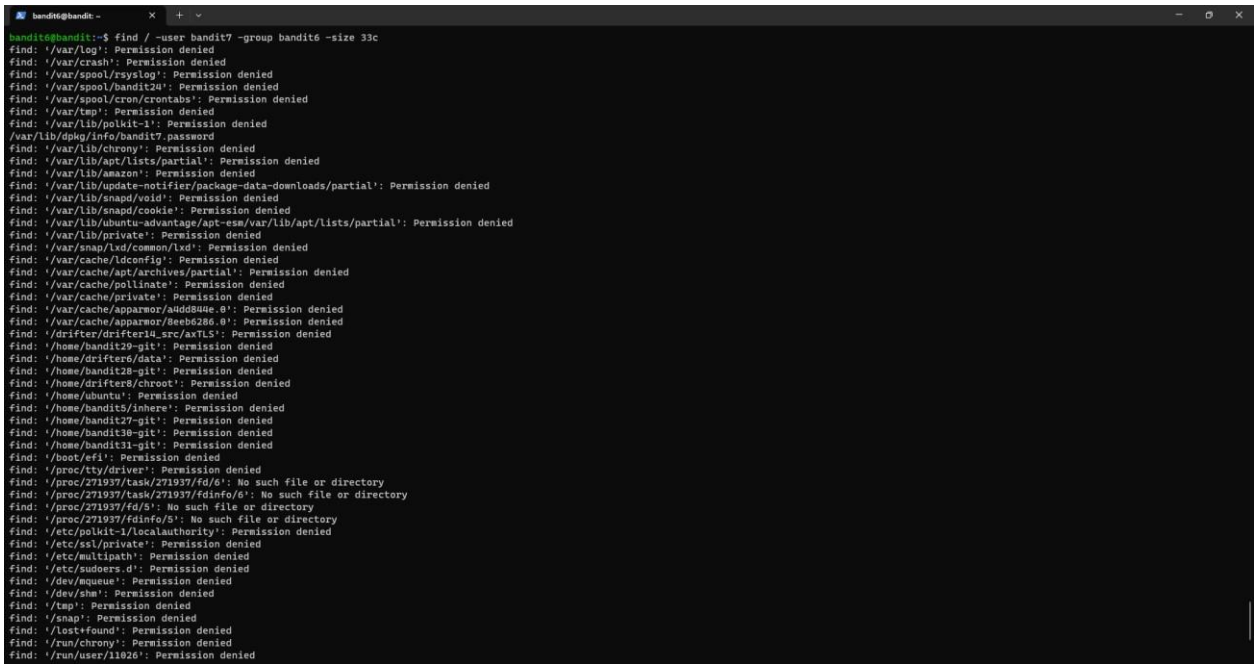
For get the password, type “cat .maybehere07/.file2”



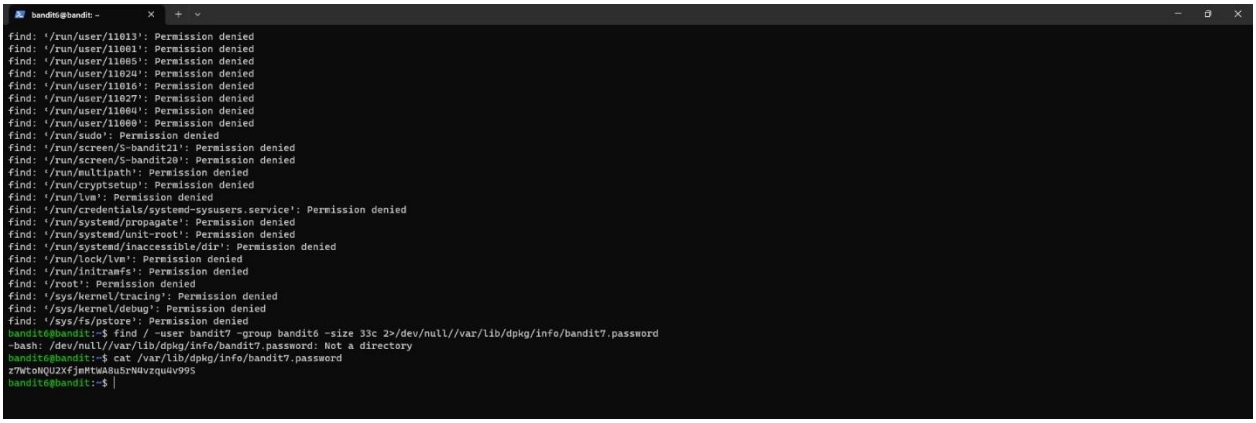
Bandit6 -> Bandit7



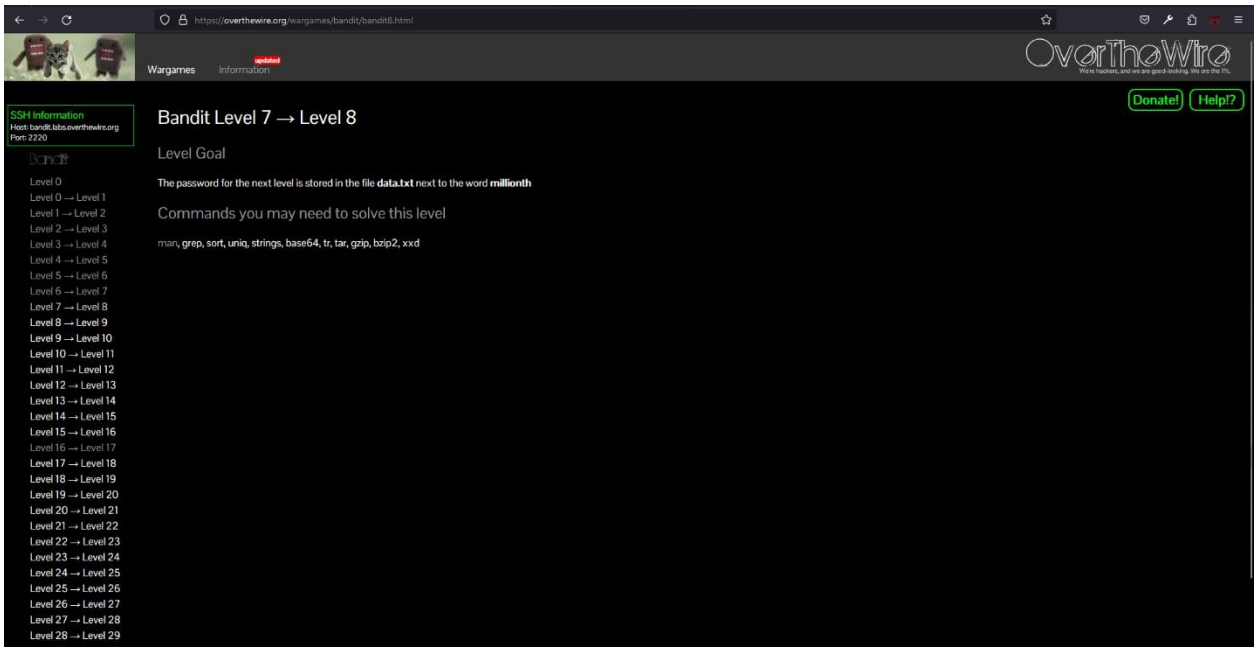
Log into the Bandit6 using the password. Use the root directory command to search the system.



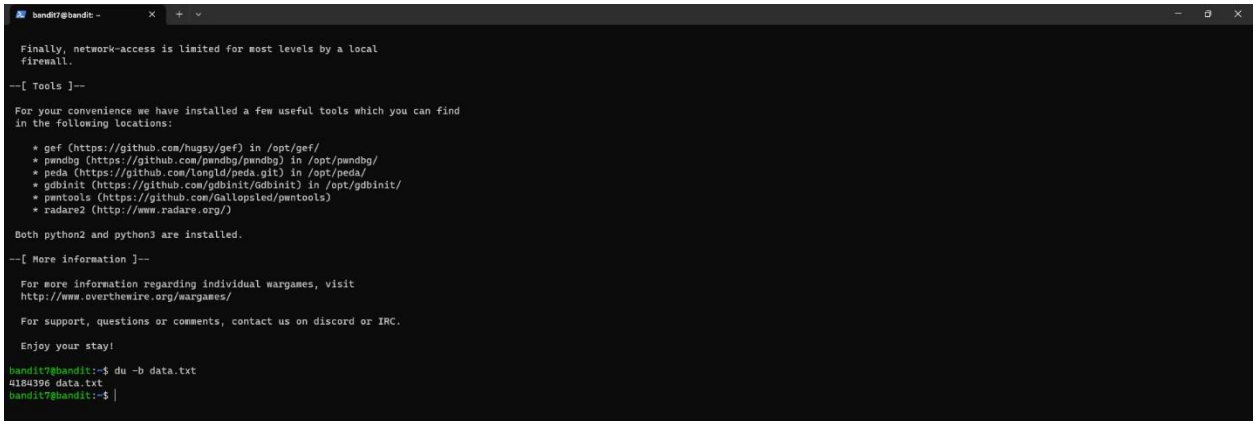
Type this command “cat /var/lib/dpkg/info/bandit7.password” and find the password.



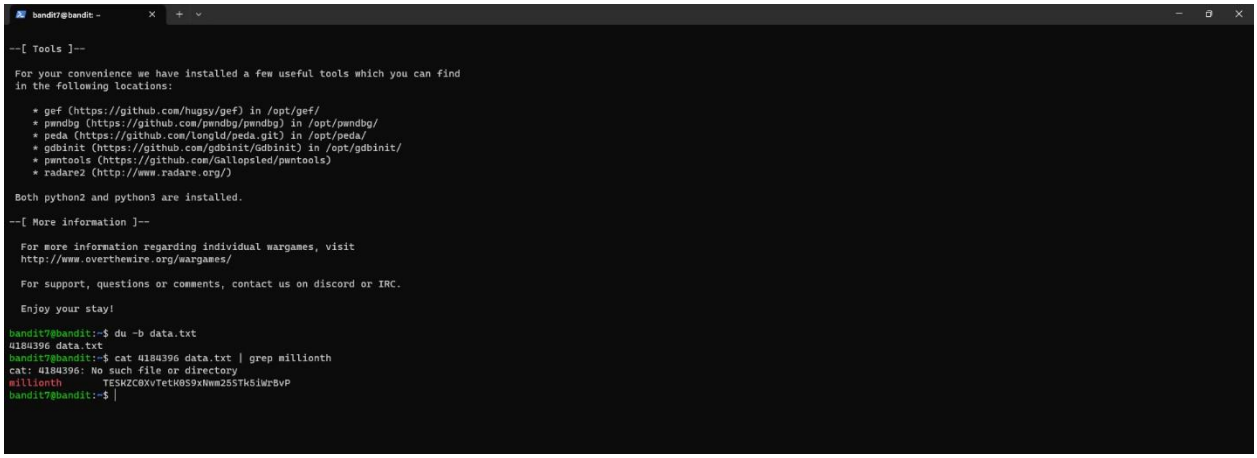
Bandit7 -> Bandit8



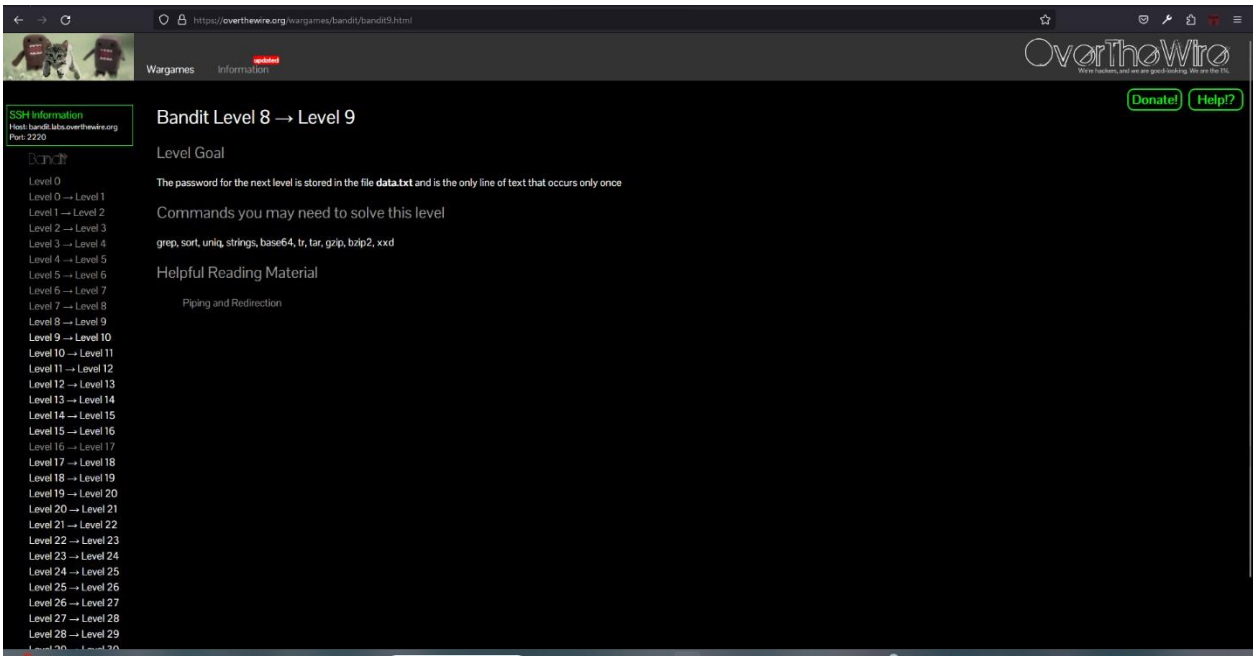
Log into Bandit7 using the password and first check the size of the “data.txt” file.



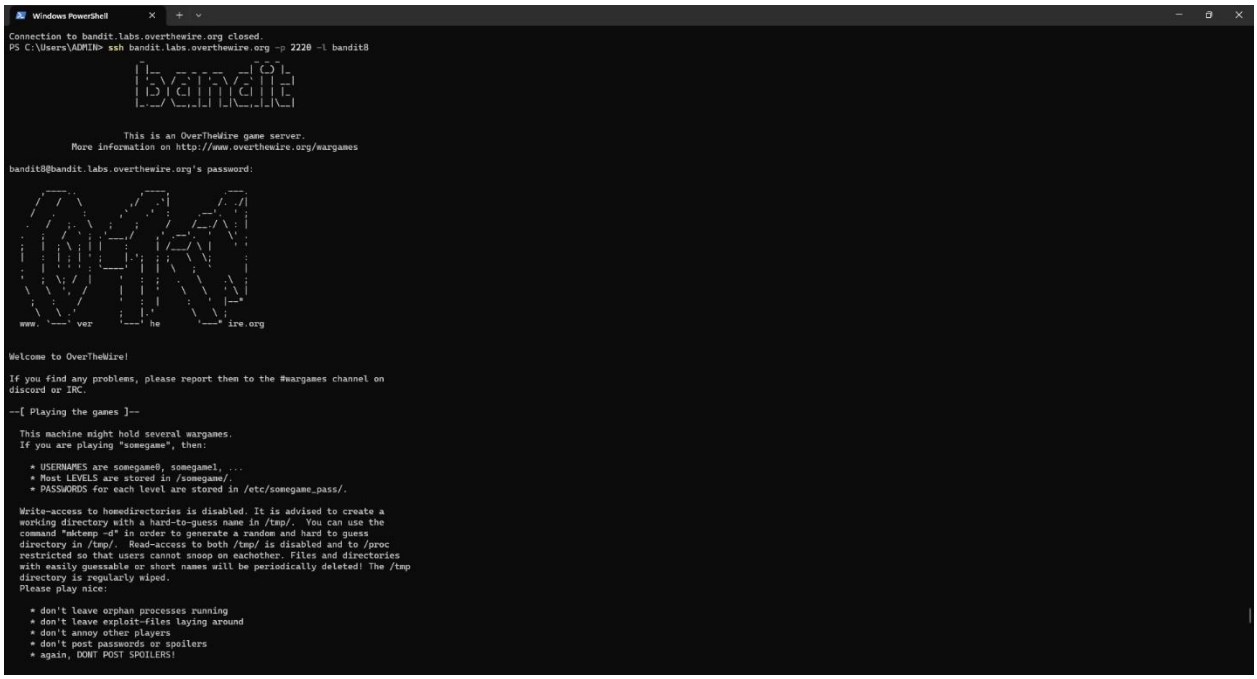
Now we need to use the “grep” command. grep command can be used to search lines that follow a particular pattern. Using the “grep” command and the pipe“|” we can find the password.



Bandit8 -> Bandit9



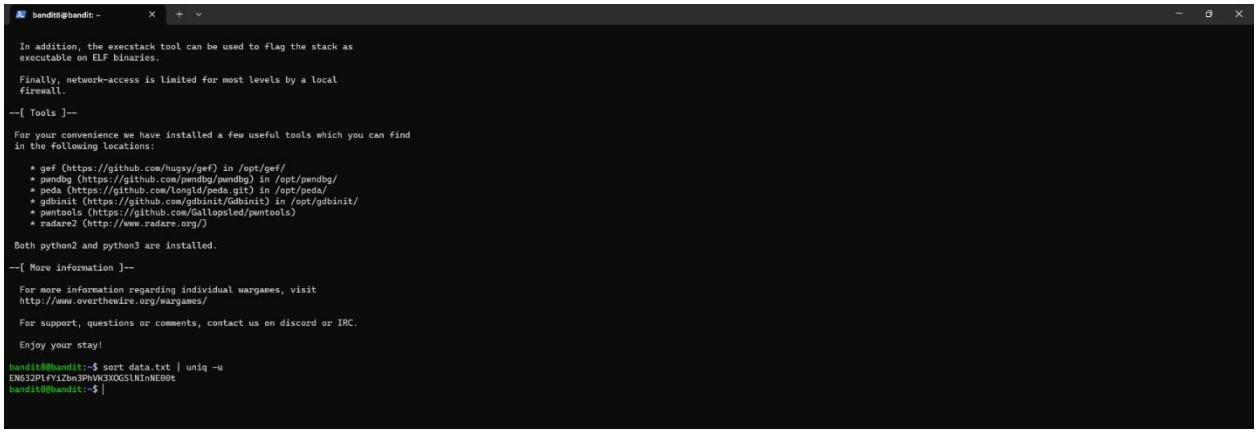
Log into Bandit8 using the password.



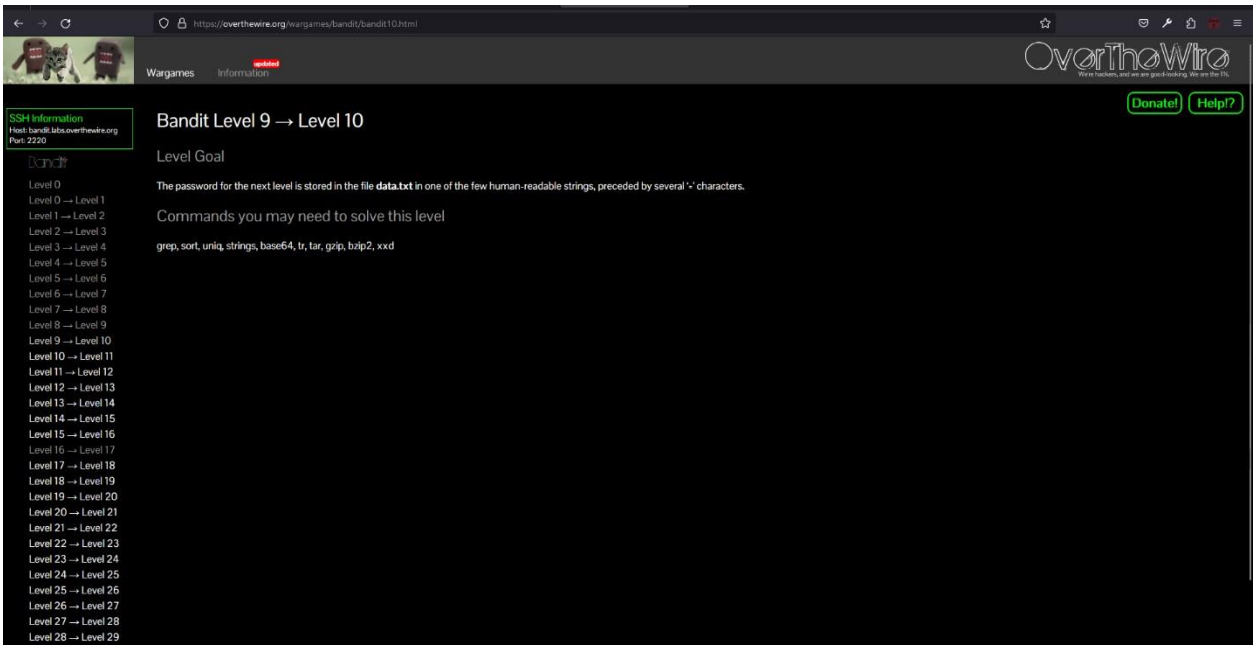
Sort – sorts the lines of a text file

Uniq – filters input and writers to the output

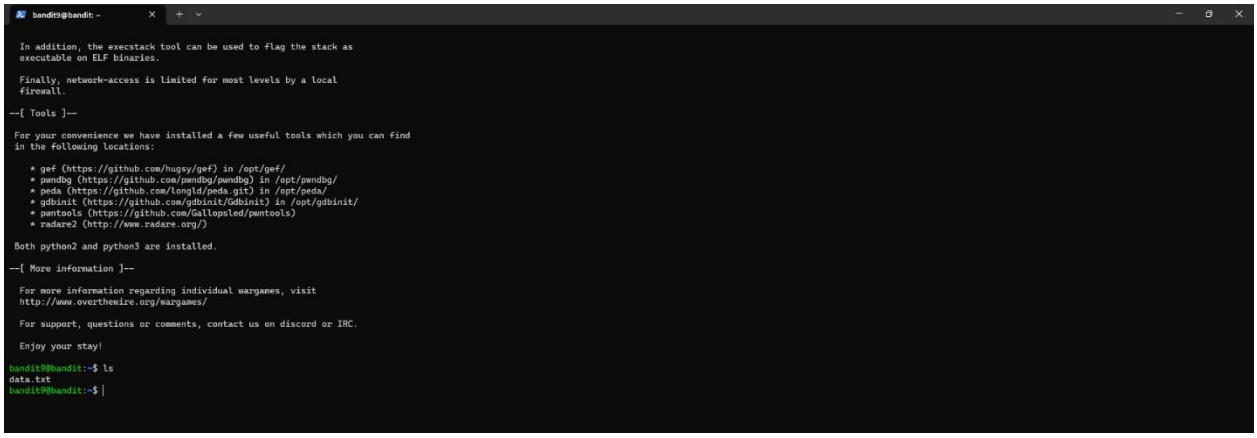
So, using “sort data.txt | uniq -u” we can get the password.



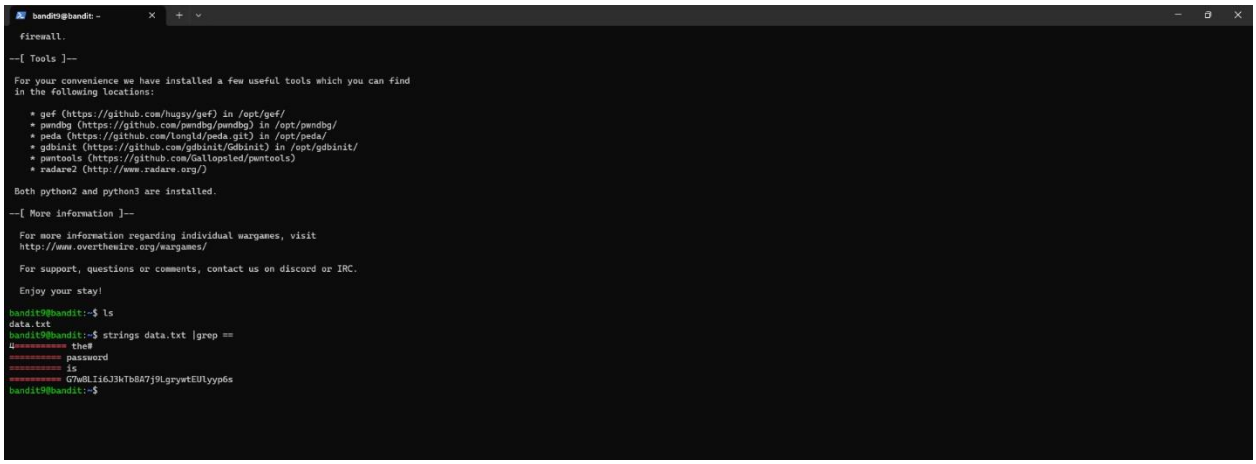
Bandit9 -> Bandit10



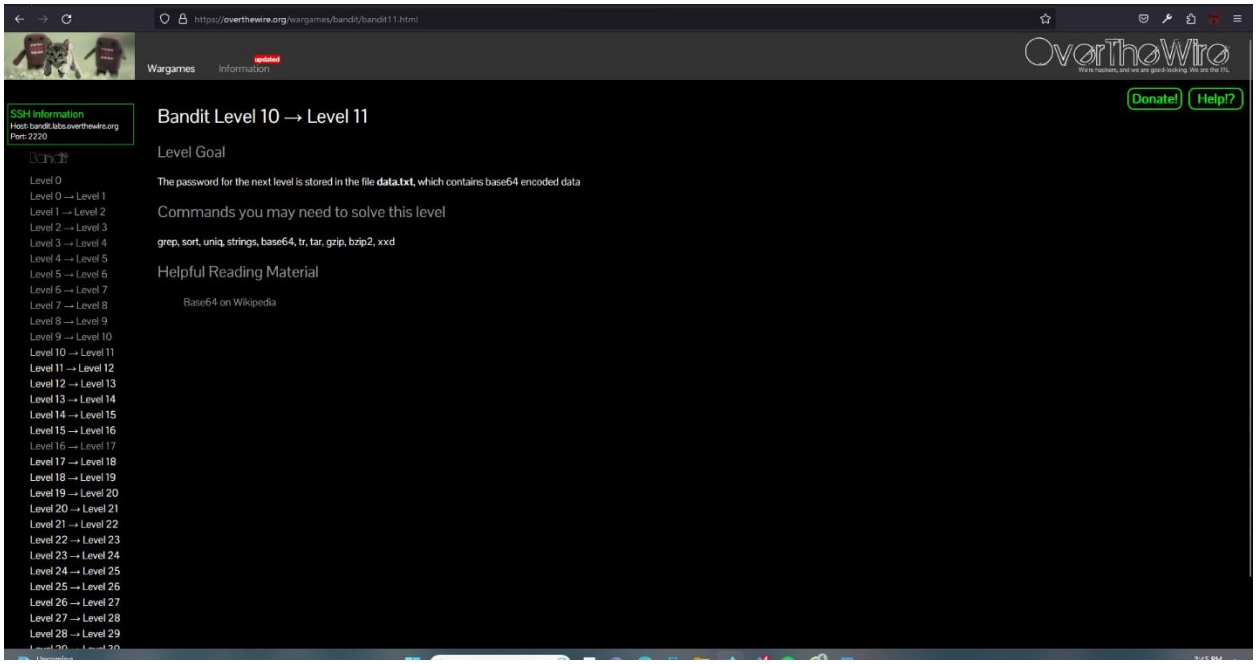
Using the “ls” command find the “data.txt” file.



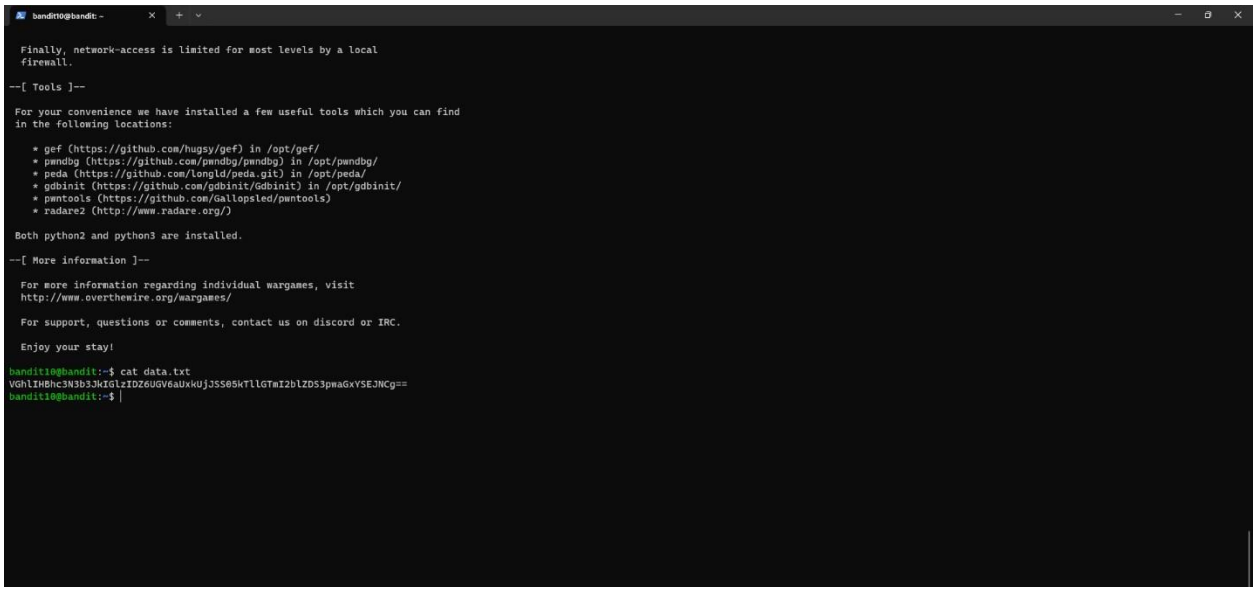
We need to use the “string” command to separate human-readable strings in “data.txt”. And use “grep” within the equal sign “=”.



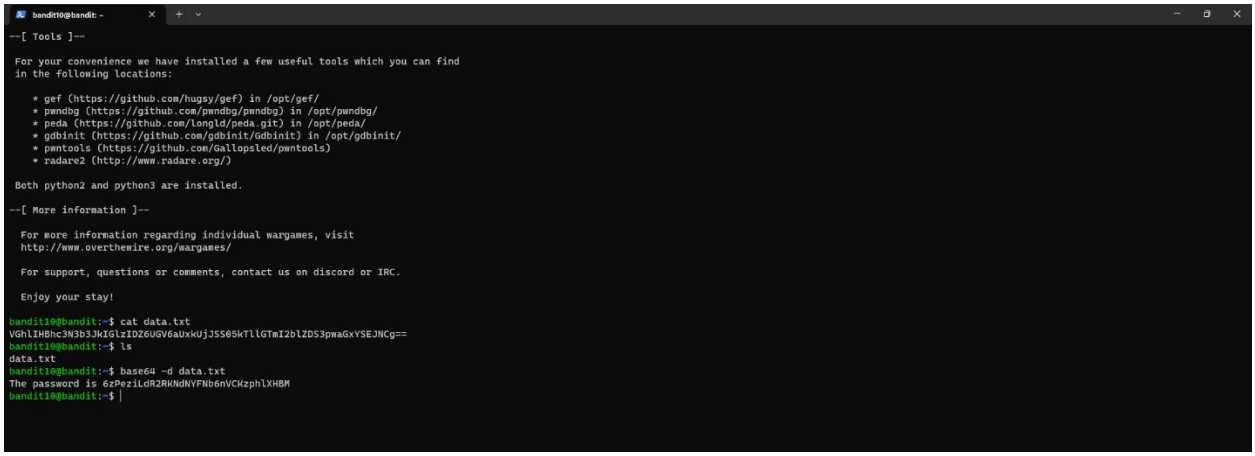
Bandit10 -> Bandit11



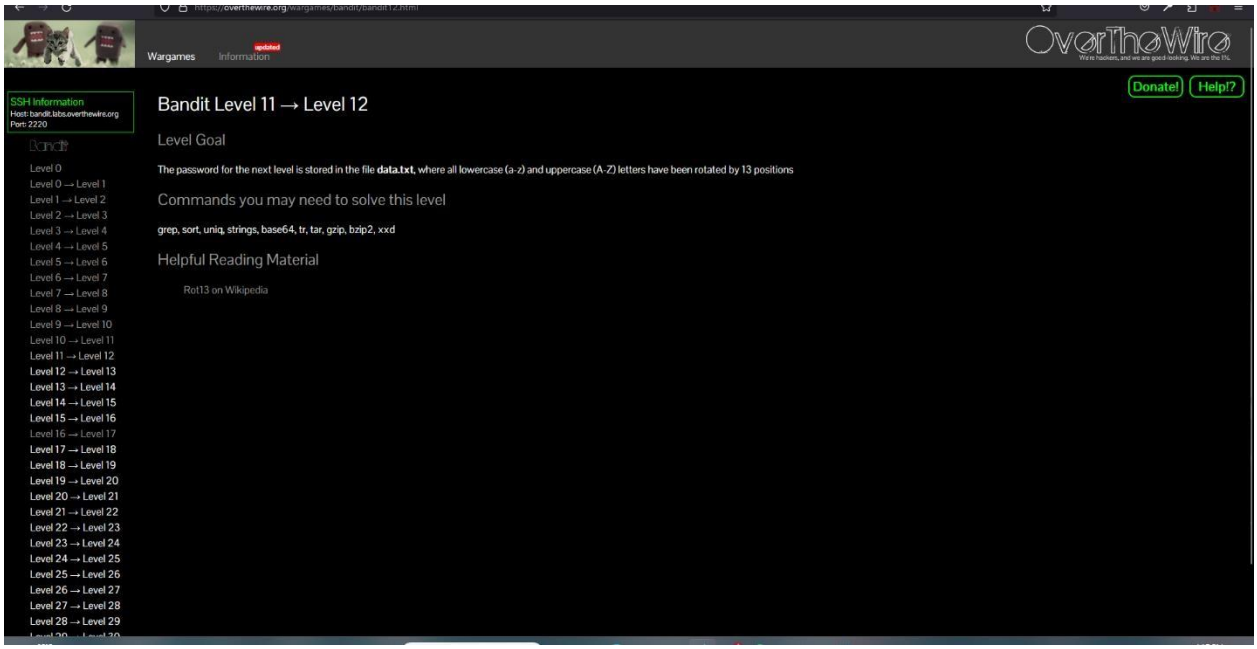
First, log into Bandit10. Run the “cat” command with the file name.



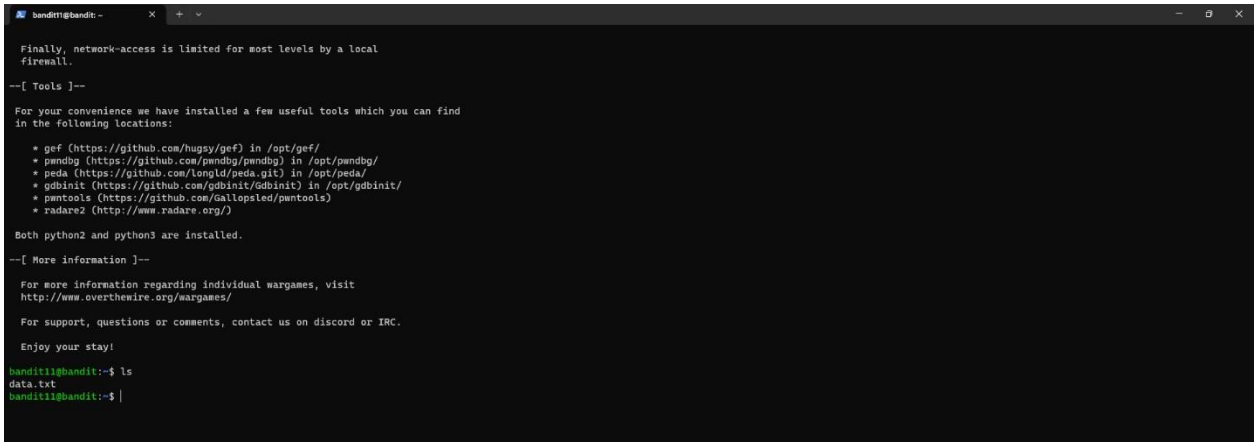
Use the “base64 -d data.txt” command for decoding to the password.



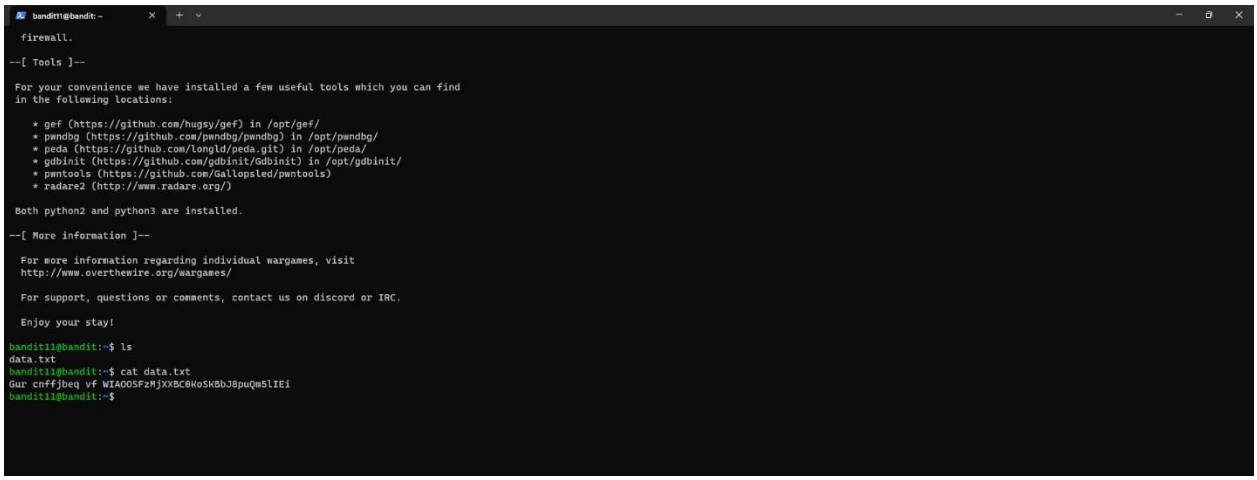
Bandit11 -> Bandit12



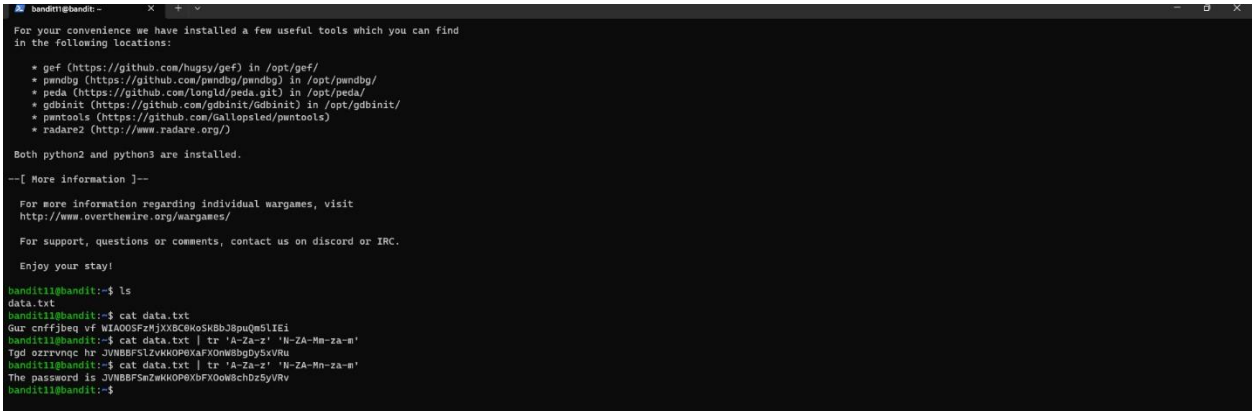
Log into Bandit11 with the password. Use the “ls” command.



Use the “cat” command to get the password.

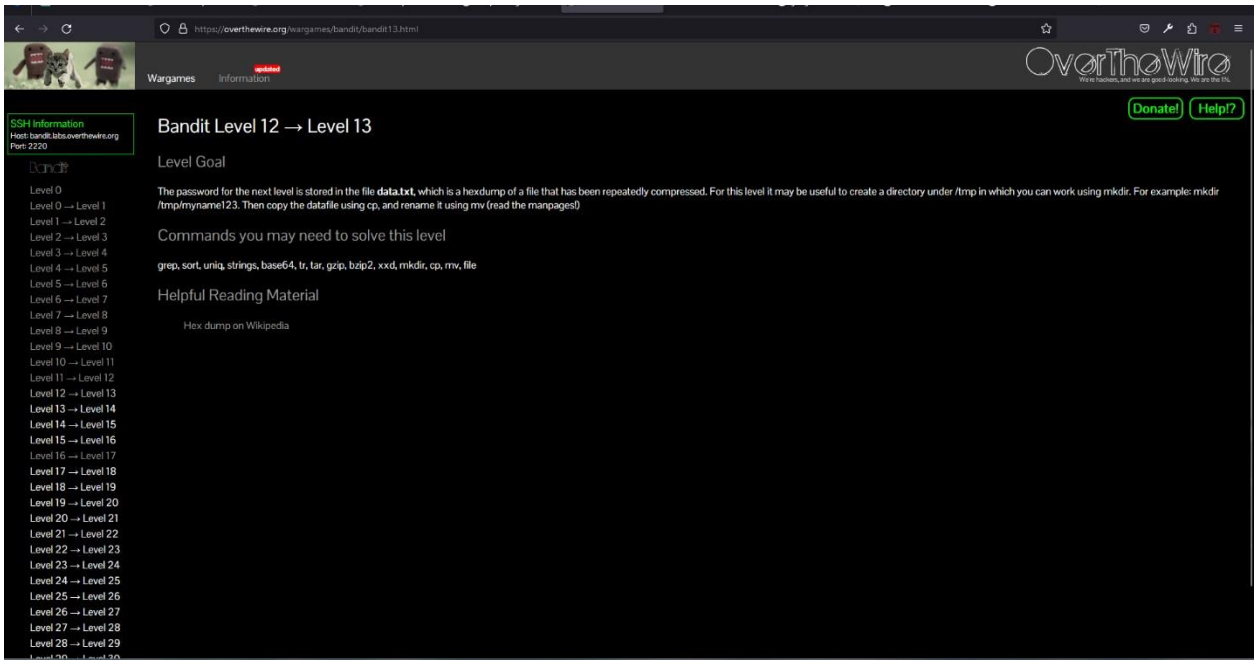


Now use the “tr” command for translation, allowing replacing the characters with others. And “A ->N, …., Z ->M” to get the password.

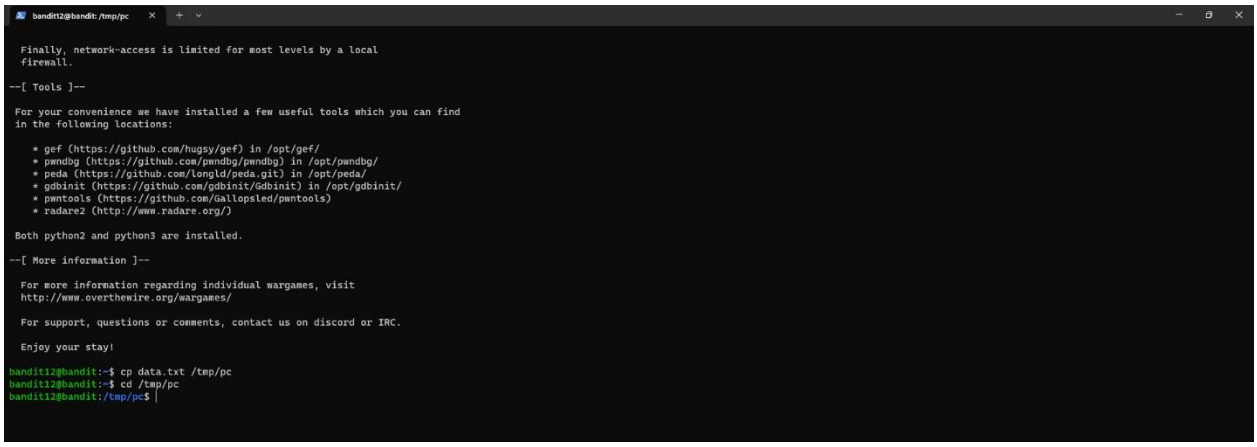


Bandit12 -> Bandit13

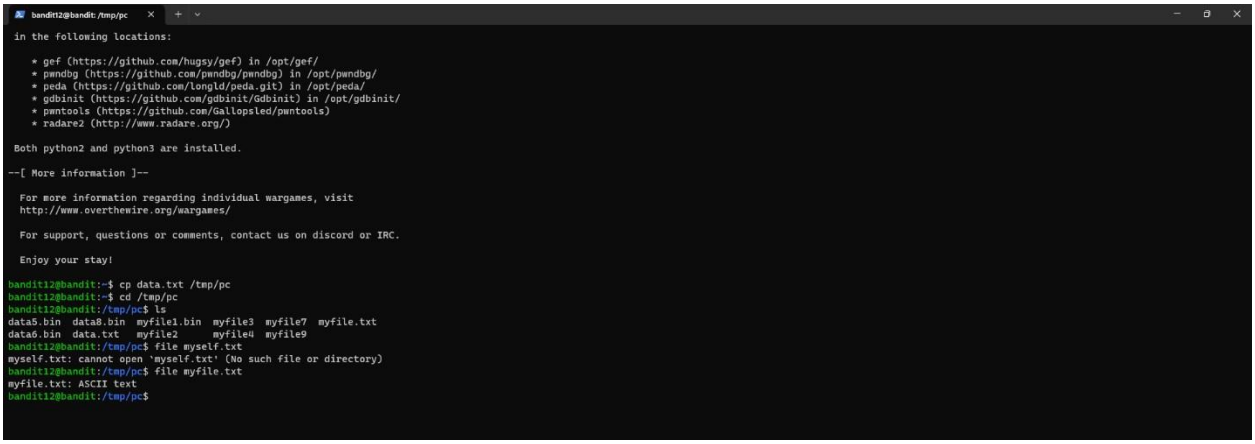
Log into Bandit12 using the password.



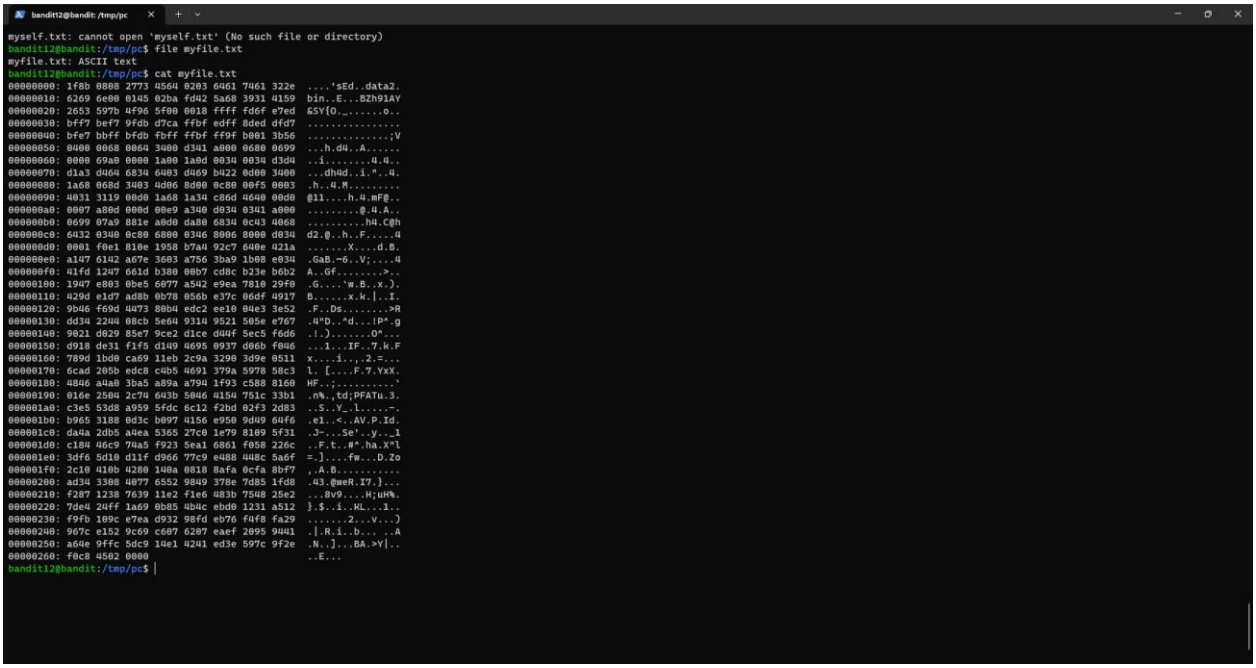
Use the “cp” command to copy files. Type “cp data.txt /tmp/pc” first and next type “cd /tmp/pc” to change directory.



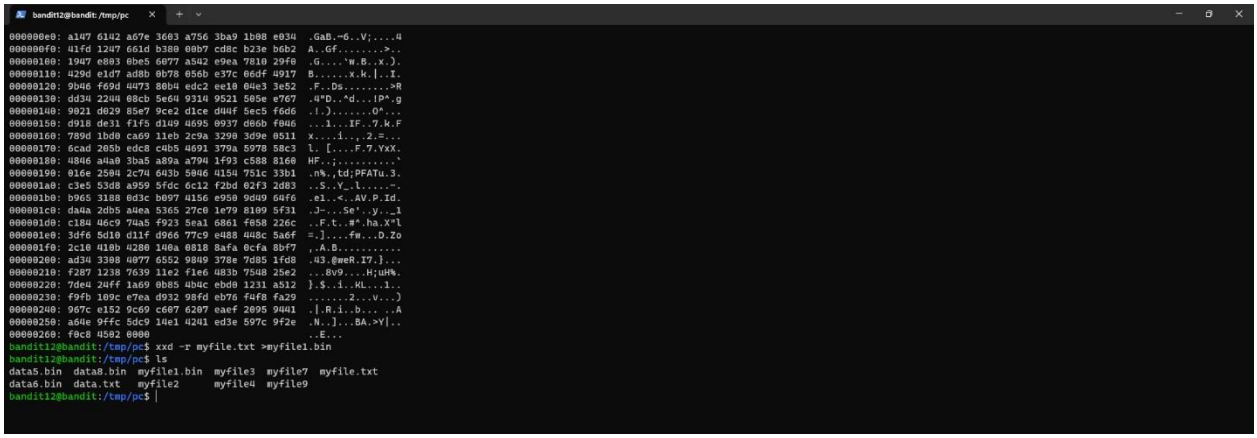
Type “ls” and find the list. Use the “file myfile.txt” and find the file to know what the password is.



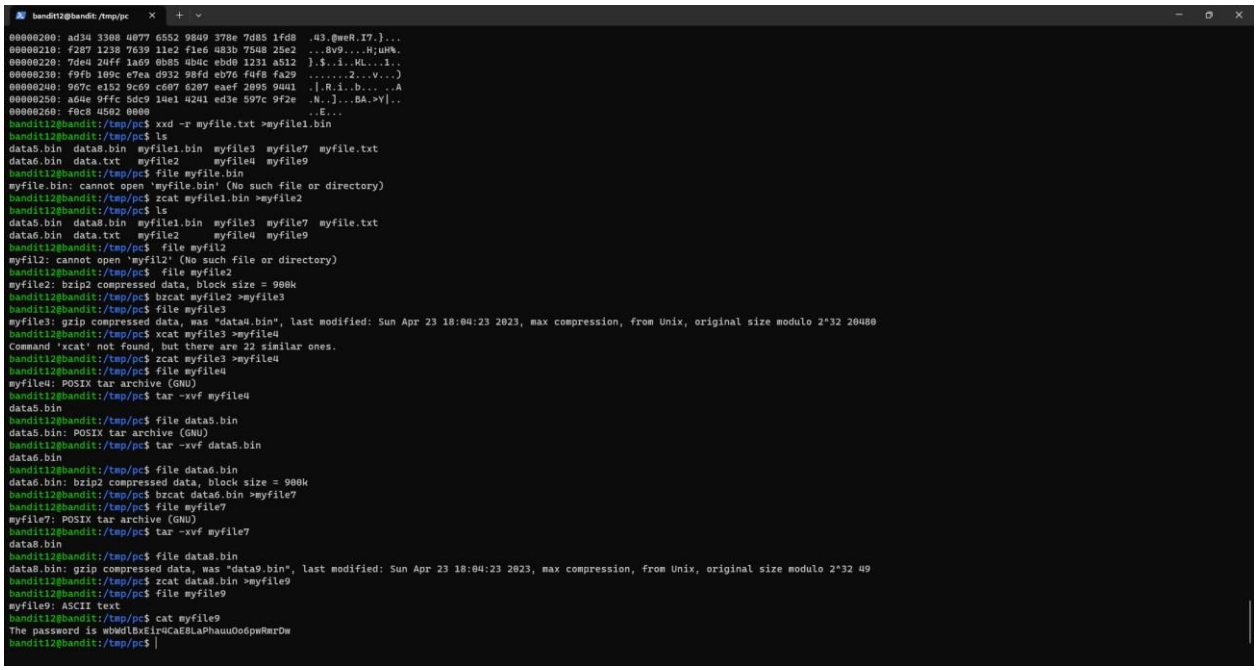
Run the “cat myfile.txt”



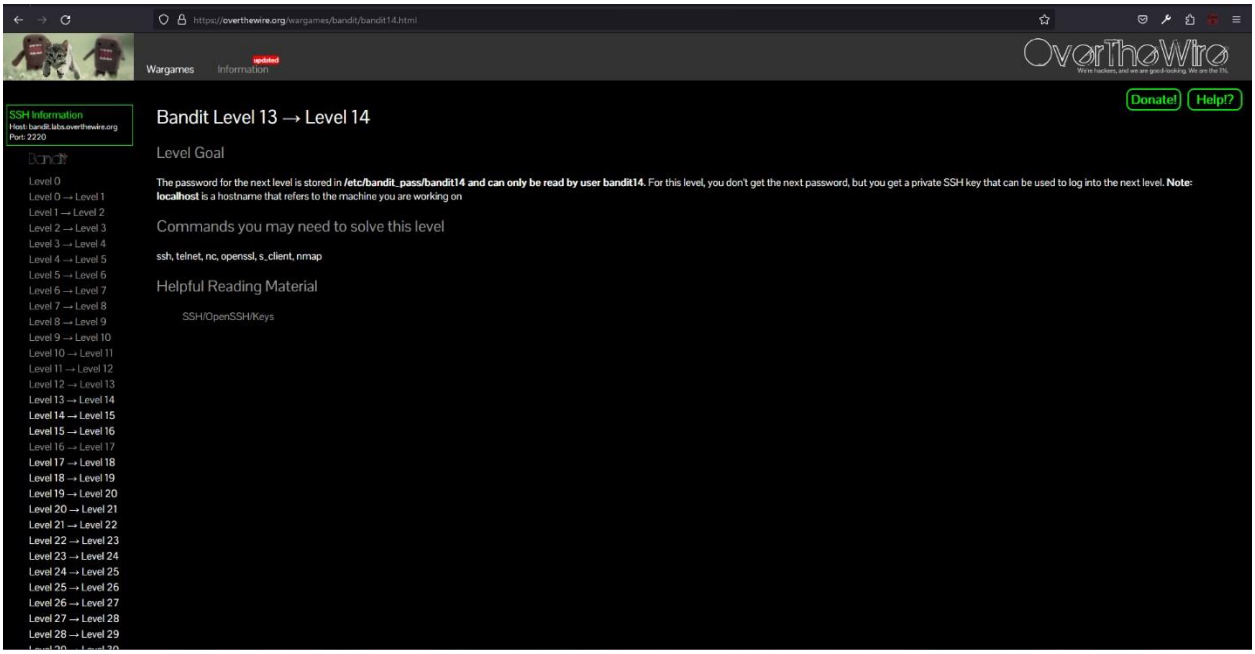
Run “xxd -r myfile.txt >myfile1.bin” and next run the “ls” to find all the files.



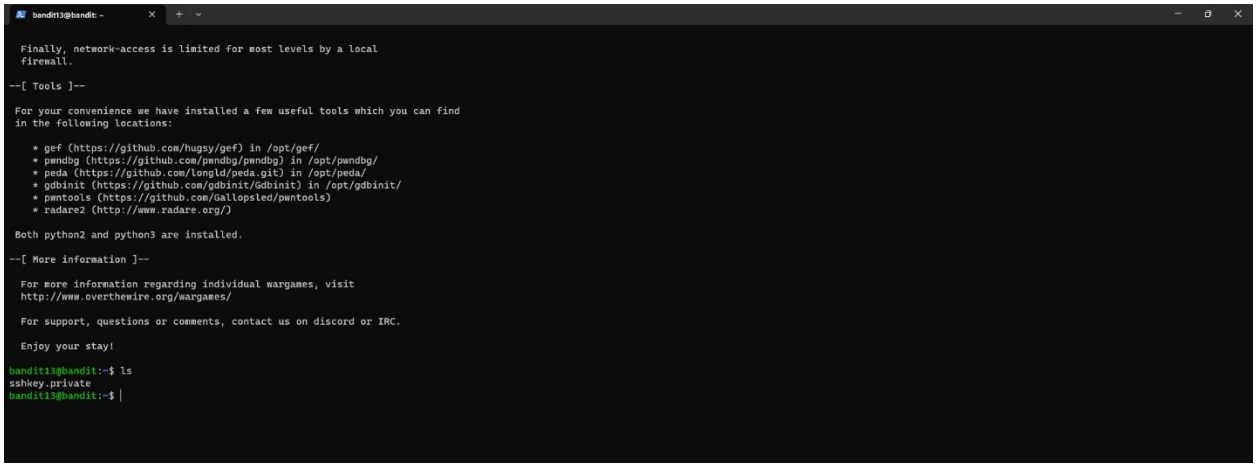
A command called “zcat” is included with “gzip” and is used to decompress “gzip” compressed files. Using the file command on myfile2, we can find bzip2 compressed data. Use that command to all 9 files and use the “tar” for archiving files and options. Finally, we can find the password.



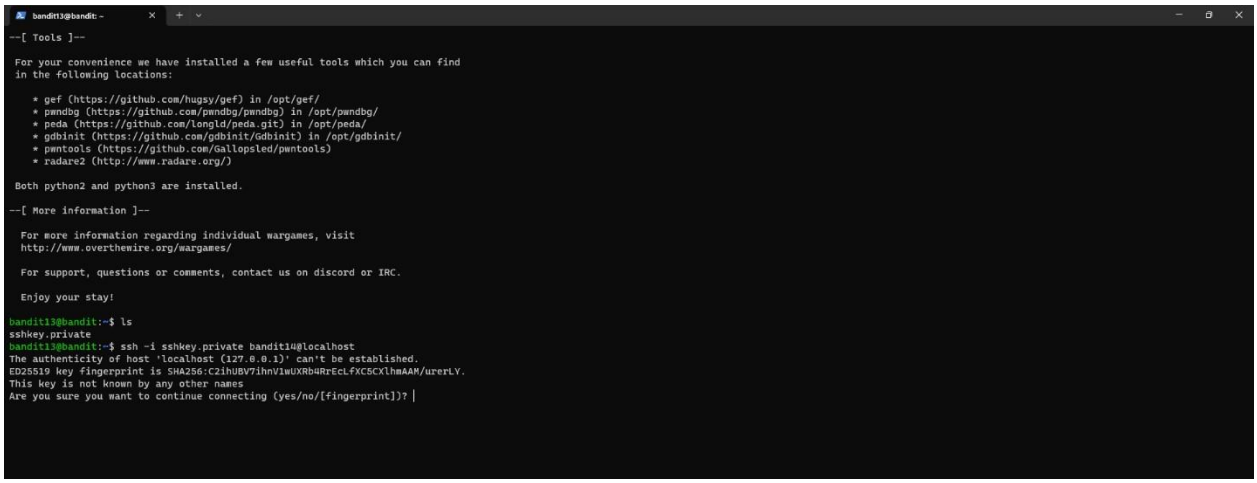
Bandit13 -> Bandit14



Use the “ls” command to find the file.

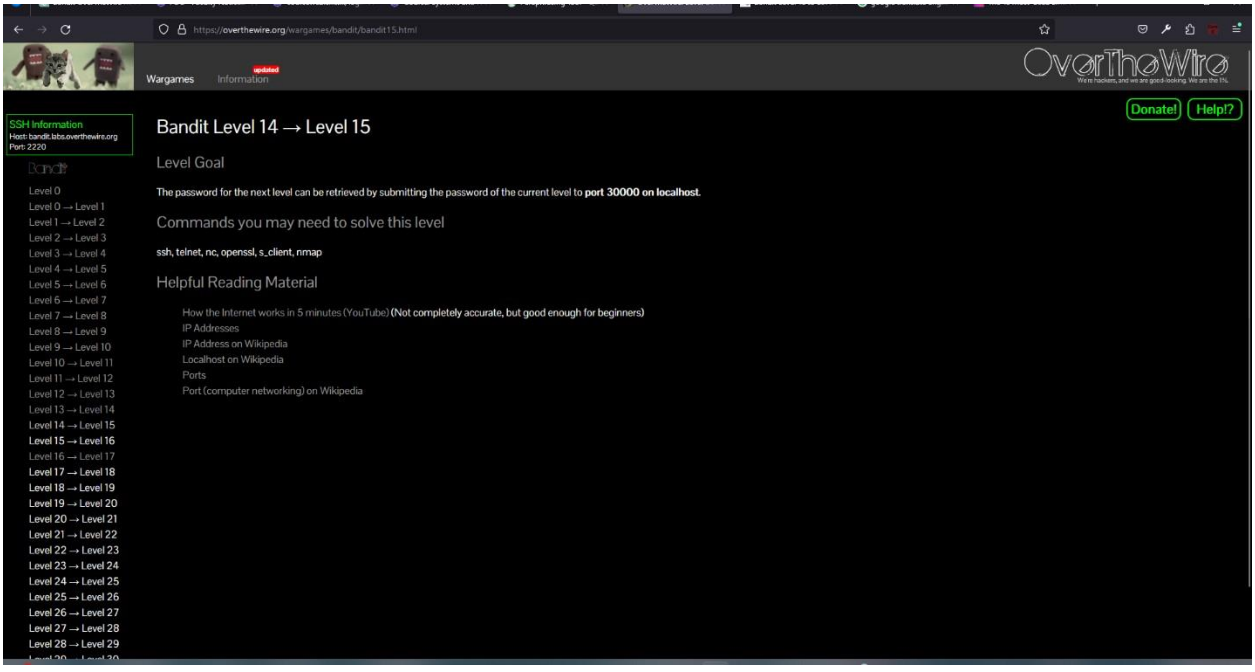


For remote machine access and command execution, use the “ssh” command. The “sshkey.private” file and the option “-i” are used to choose the identified file for RSA or DSA authentication.

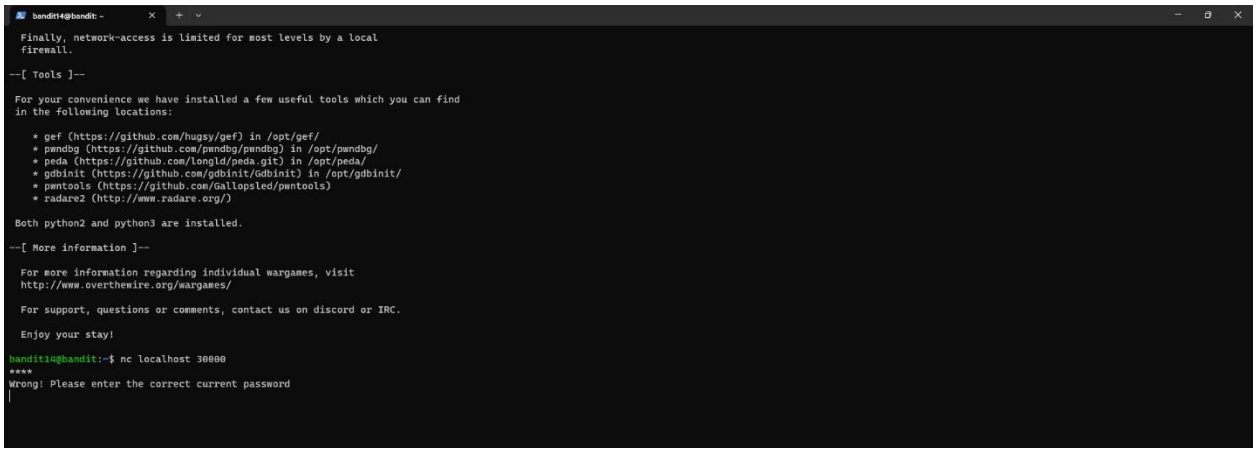


Bandit14 -> Bandit15

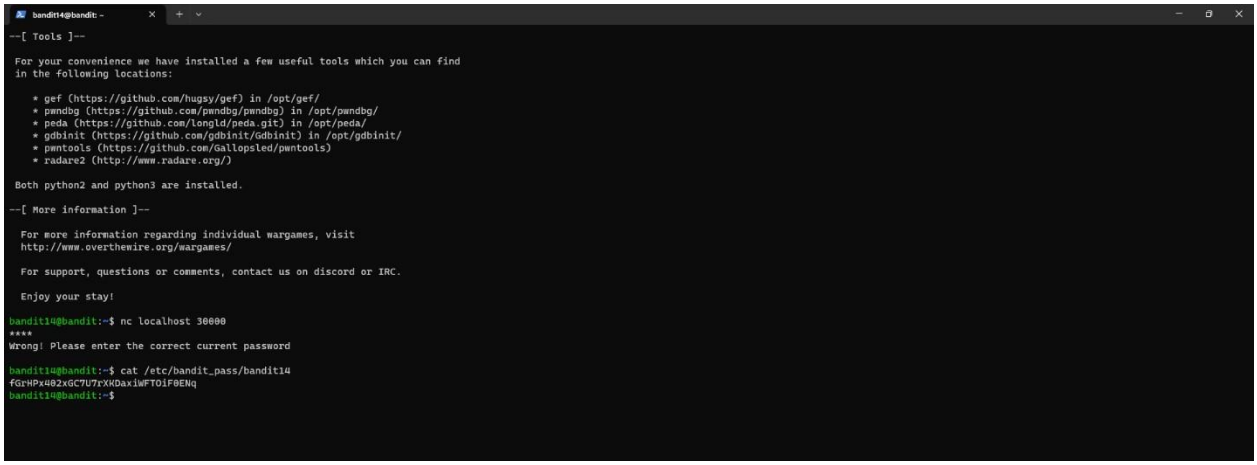
Log into Bandit14 first.

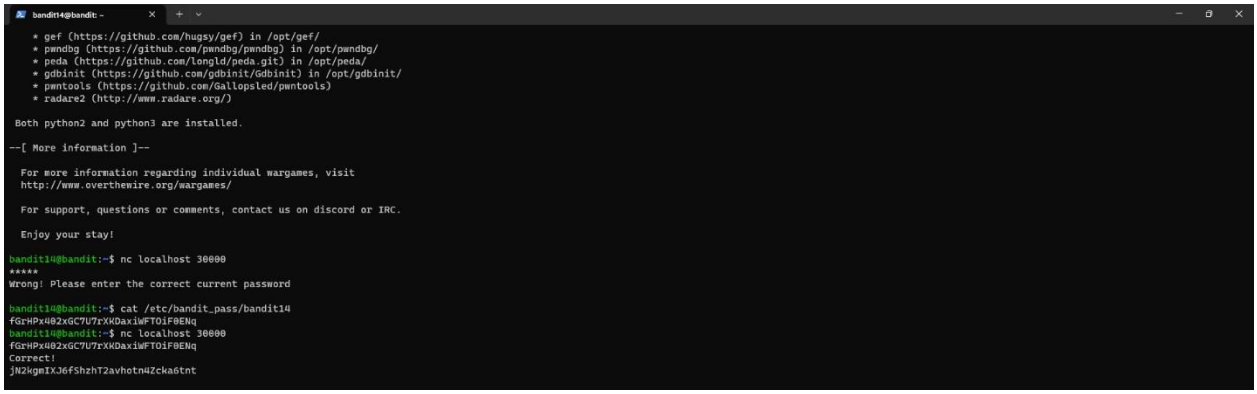


The command “nc” enables the reading and writing of data across a network connection. Both TCP and UDP connections are supported by it. Use that command and try to get the password. But they asked us for the password. So we need to find the password first.

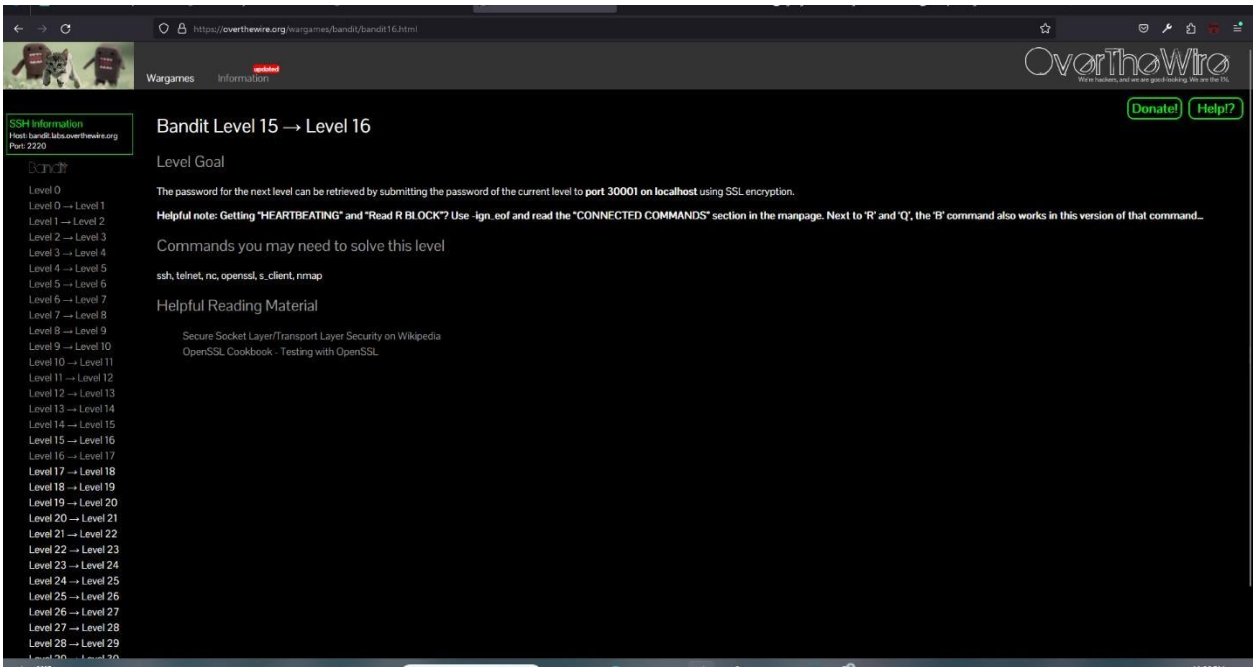


Run this command “cat /etc/bandit_pass/bandit14” and get the password.





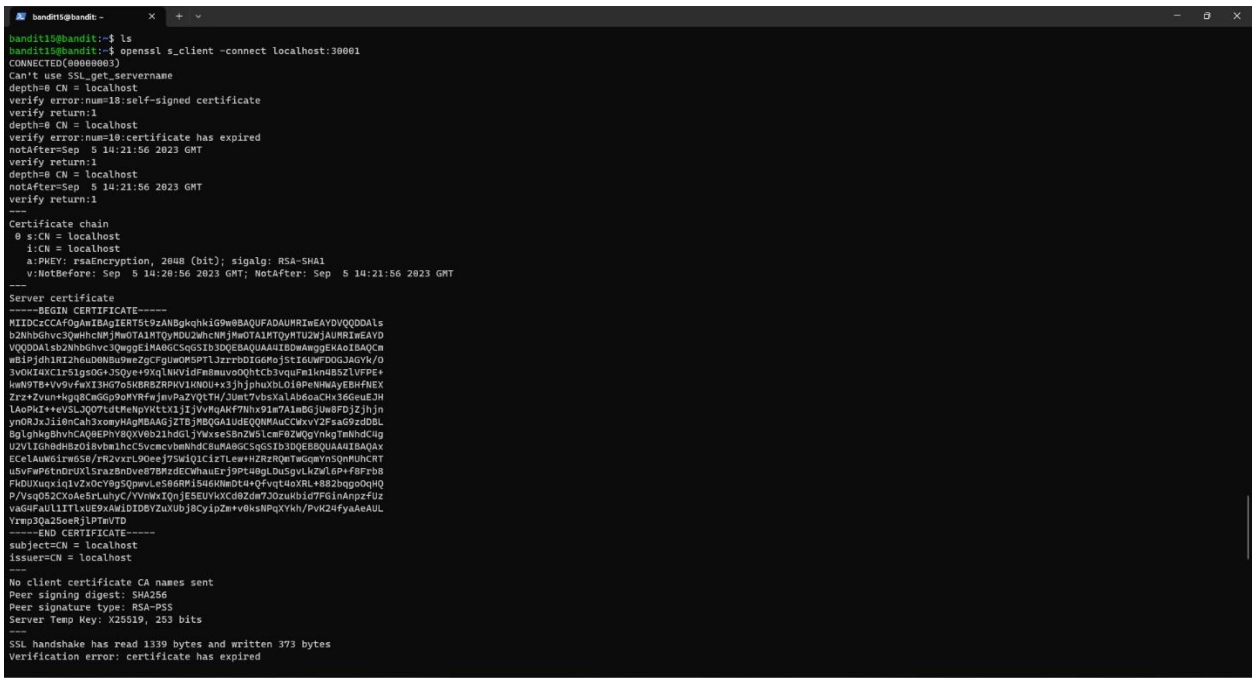
Bandit15 -> Bandit16

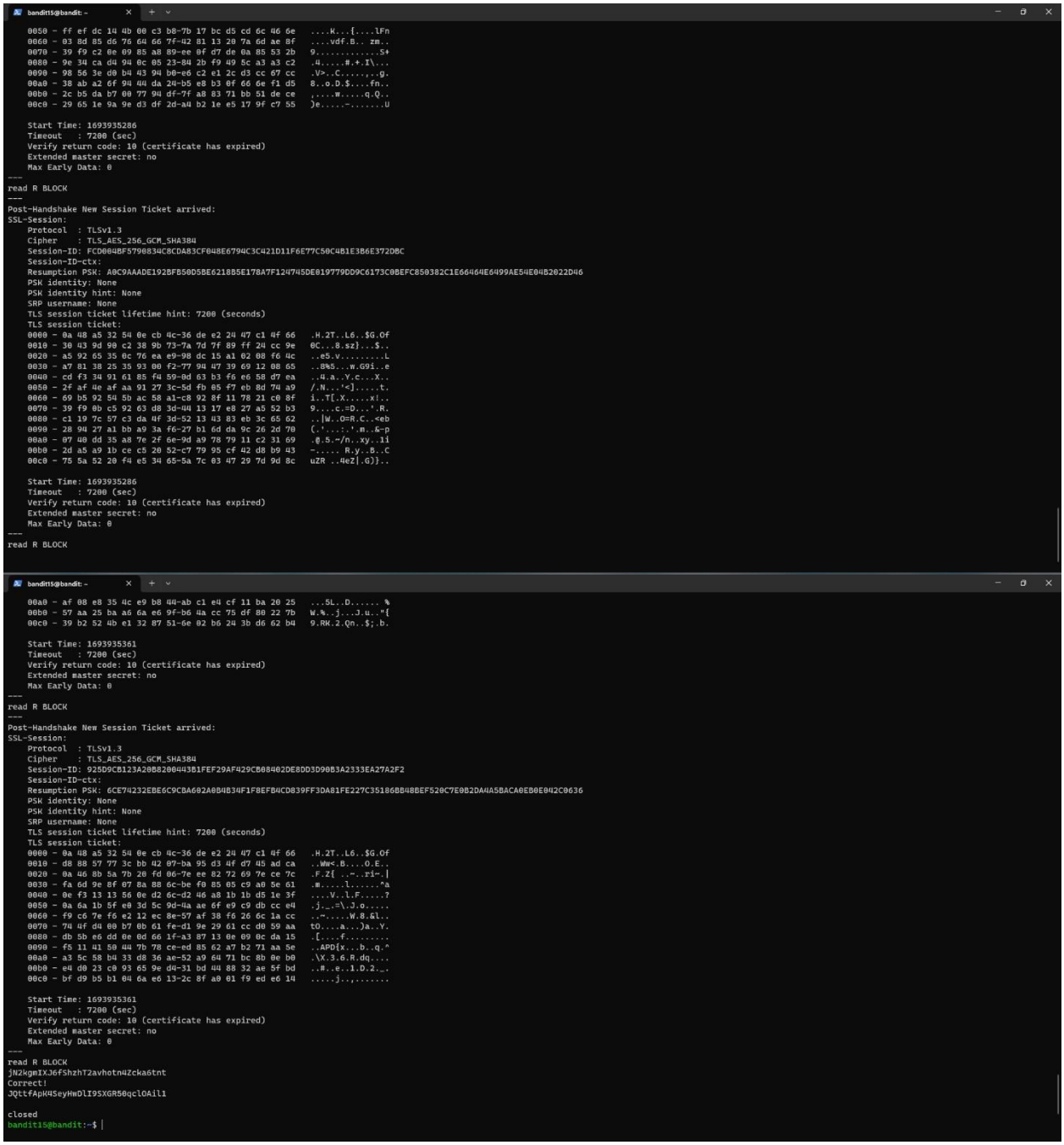


Openssl – library for secure communication over networks.

Openssl s_client – implementation of a basic SSl/TLS client that communication with a server.

Run “openssl s_client -connect localhost:30001” this command, after run that we need to type the password of bandit 15. When we type it shows the Bandit16 password.



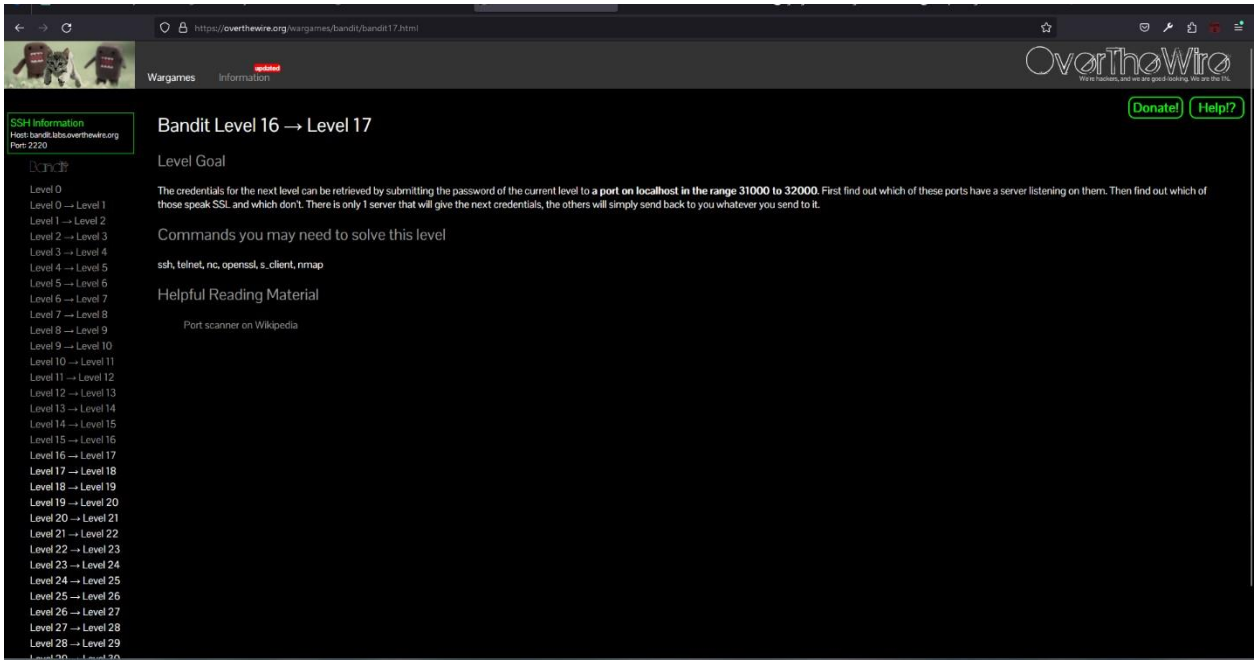

bandit15@bandit: ~
0050 - ff ef dc 14 4b 00 c3 b8-7b 17 bc d5 cd 6c 46 6e ....K...{....lFn
0060 - 03 8d 85 d6 76 64 66 7f-42 81 13 20 7a 6d ae 8f ....vdf.B.. zm..
0070 - 39 f9 c2 0e 09 85 a8 89-ee 0f d7 de 0a 85 53 2b 9.............S+
0080 - 9e 34 ca d4 94 0c 05 23-84 2b f9 49 5c a3 a3 c2 .4.....#.+.I\...
0090 - 98 56 3e d0 b4 43 94 b0-e6 c2 e1 2c d3 cc 67 cc .V>..C.....,..g.
00a0 - 38 ab a2 6f 94 44 da 24-b5 e8 b3 0f 66 6e f1 d5 8..o.D.$....fn..
00b0 - 2c b5 da b7 00 77 94 df-7f a8 83 71 bb 51 de ce ,....w.....q.Q..
00c0 - 29 65 1e 9a 9e d3 df 2d-a4 b2 1e e5 17 9f c7 55)e.....-.......U
Start Time: 1693935286
Timeout : 7200 (sec)
Verify return code: 10 (certificate has expired)
Extended master secret: no
Max Early Data: 0
---
read R BLOCK

Post-Handshake New Session Ticket arrived:
SSL-Session:
Protocol : TLSv1.3
Cipher : TLS_AES_256_GCM_SHA384
Session-ID: FCD804BF5790834C8CDA83CF048E6794C3C421D11F6E77C50C4B1E3B6E372DBC
Session-ID-ctx:
Resumption PSK: A0C9AAADE192BF850D5BE6218B5E178A7F124745DE019779DD9C6173C0BEFC850382C1E66464E6499AE54E04B2022D46
PSK identity: None
PSK identity hint: None
SRP username: None
TLS session ticket lifetime hint: 7200 (seconds)
TLS session ticket:
0000 - 0a 48 a5 32 54 0e cb 4c-36 de e2 24 47 c1 4f 66 .H.2T..L6..$G.Of
0010 - 30 43 9d 90 c2 38 9b 73-7a 7d 7f 89 ff 24 cc 9e 0C...8.sz}...$..
0020 - a5 92 65 35 0c 76 ea e9-98 dc 15 a1 02 08 f6 4c ..e5.v.........L
0030 - a7 81 38 25 35 93 00 f2-77 94 47 39 69 12 08 65 ..8%5...w.G9i..e
0040 - cd f3 34 91 61 85 f4 59-0d 63 b3 f6 e6 58 d7 ea ..4.a..Y.c...X..
0050 - 2f af 4e af aa 91 27 3c-5d fb 05 f7 eb 8d 74 a9 /.N...'<]....t.
0060 - 69 b5 92 54 5b ac 58 a1-c8 92 8f 11 78 21 c0 8f i..T[.X.....x!..
0070 - 39 f0 0b c5 92 63 d8 3d-44 13 17 e8 27 a5 52 b3 9....c.=D...'.R.
0080 - c1 19 7c 57 c3 da 4f 3d-52 13 43 83 eb 3c 65 62 ..|W..O=R.C..<eb
0090 - 28 94 27 a1 bb a9 3a f6-27 b1 6d da 9c 26 2d 70 (.'...:.'.m..&-p
00a0 - 07 40 dd 35 a8 7e 2f 6e-9d a9 78 79 11 c2 31 69 .@.5.~/n..xy..1i
00b0 - 2d a5 a9 1b ce c5 20 52-c7 79 95 cf 42 d8 b9 43 -..... R.y..B..C
00c0 - 75 5a 52 20 f4 e5 34 65-5a 7c 03 47 29 7d 9d 8c uZR ..4eZ|.G)}..
Start Time: 1693935286
Timeout : 7200 (sec)
Verify return code: 10 (certificate has expired)
Extended master secret: no
Max Early Data: 0
---
read R BLOCK
bandit15@bandit: ~
00a0 - af 08 e8 35 4c e9 b8 44-ab c1 e4 cf 11 ba 20 25 ...5L..D..... %
00b0 - 57 aa 25 ba a6 6a e6 9f-b6 4a cc 75 df 80 22 7b W.%..j...J.u..."{
00c0 - 39 b2 52 4b e1 32 87 51-6e 02 b6 24 3b d6 62 b4 9.RK.2.Qn..$;.b.
Start Time: 1693935361
Timeout : 7200 (sec)
Verify return code: 10 (certificate has expired)
Extended master secret: no
Max Early Data: 0
---
read R BLOCK

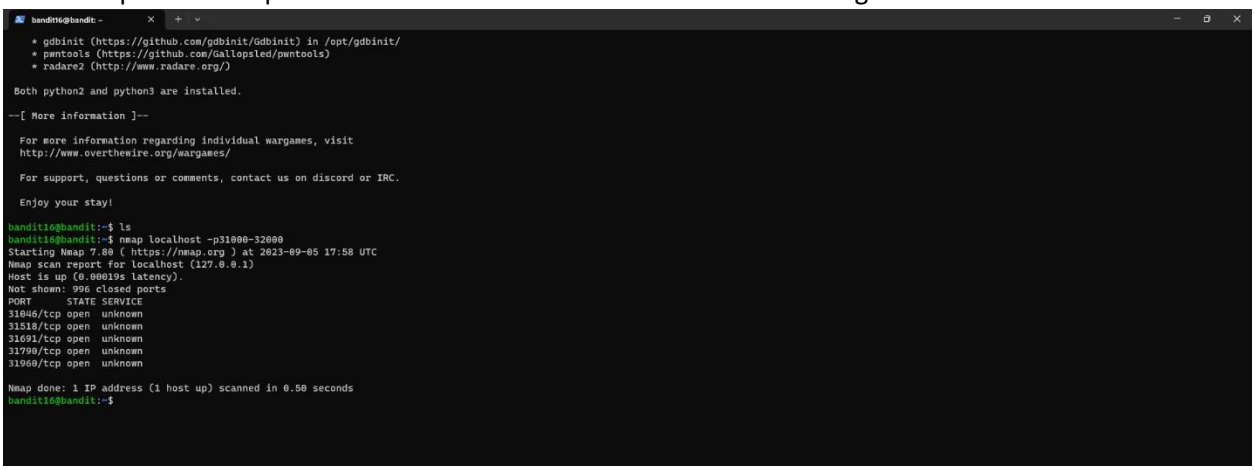
Post-Handshake New Session Ticket arrived:
SSL-Session:
Protocol : TLSv1.3
Cipher : TLS_AES_256_GCM_SHA384
Session-ID: 925D9CB123A20B82004433B1FEF29AF429CB08402DE8DD3D90B3A2333EA27A2F2
Session-ID-ctx:
Resumption PSK: 6CE74232EBE6C9CBA602A0B4B34F1F8EFB4CD839FF3DA81FE227C35186BB48BEF520C7E0B2DA4A5BACA0EB0E042C0636
PSK identity: None
PSK identity hint: None
SRP username: None
TLS session ticket lifetime hint: 7200 (seconds)
TLS session ticket:
0000 - 0a 48 a5 32 54 0e cb 4c-36 de e2 24 47 c1 4f 66 .H.2T..L6..$G.Of
0010 - d8 88 57 77 3c bb 42 07-ba 95 d3 4f d7 45 ad ca ..Ww<.B....O.E..
0020 - 0a 46 8b 5a 7b 20 fd 06-7e ee 82 72 69 7e ce 7c .F.Z{ ..~..ri~.|
0030 - fa 6d 9e 8f 07 8a 88 6c-be f0 85 05 c9 a0 5e 61 .m.....l......^a
0040 - 0e f3 13 13 56 0e d2 6c-d2 46 a8 1b 1b d5 1e 3f ....V..l.F.....?
0050 - 0a 6a 1b 5f e0 3d 5c 9d-4a ae 6f e9 c9 db cc e4 .j._.=\.J.o.....
0060 - f9 c6 7e f6 e2 12 ec 8e-57 af 38 f6 26 6c 1a cc ..~.....W.8.&l..
0070 - 74 4f d4 00 b7 0b 61 fe-d1 9e 29 61 cc d0 59 aa tO....a...)a..Y.
0080 - db 5b e6 dd 0e 0d 66 1f-a3 87 13 0e 09 0c da 15 .[....f.........
0090 - f5 11 41 50 44 7b 78 ce-ed 85 62 a7 b2 71 aa 5e ..APD{x...b..q.^
00a0 - a3 5c 58 b4 33 d8 36 ae-52 a9 64 71 bc 8b 0e b0 .\X.3.6.R.dq....
00b0 - e4 d0 23 c9 93 65 9e d4-31 bd 44 88 32 ae 5f bd ..#..e..1.D.2._.
00c0 - bf d9 b5 b1 04 6a e6 13-2c 8f a0 01 f9 ed e6 14 .....j..,.......
Start Time: 1693935361
Timeout : 7200 (sec)
Verify return code: 10 (certificate has expired)
Extended master secret: no
Max Early Data: 0
---
read R BLOCK
jN2kgmIXJ6fShzhT2avhotn4Zcka6tnt
Correct!
JQttfApK4SeyHwDlI9SXGR50qclOAil1
closed
bandit15@bandit:~$

Bandit16 -> Bandit17

Log into the Bandit16.

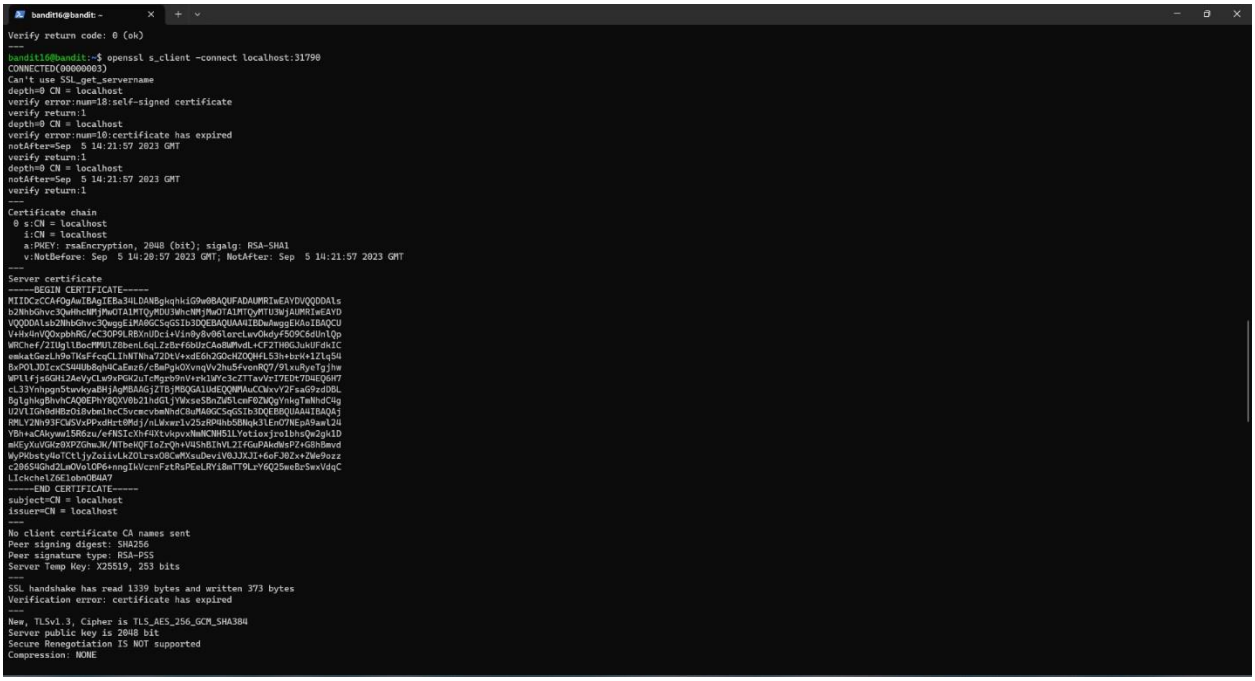


Run “nmap localhost -p31000-32000” to check what services are running on them.



The port that shows promise appears to be 31790, which is used by an unidentified service.

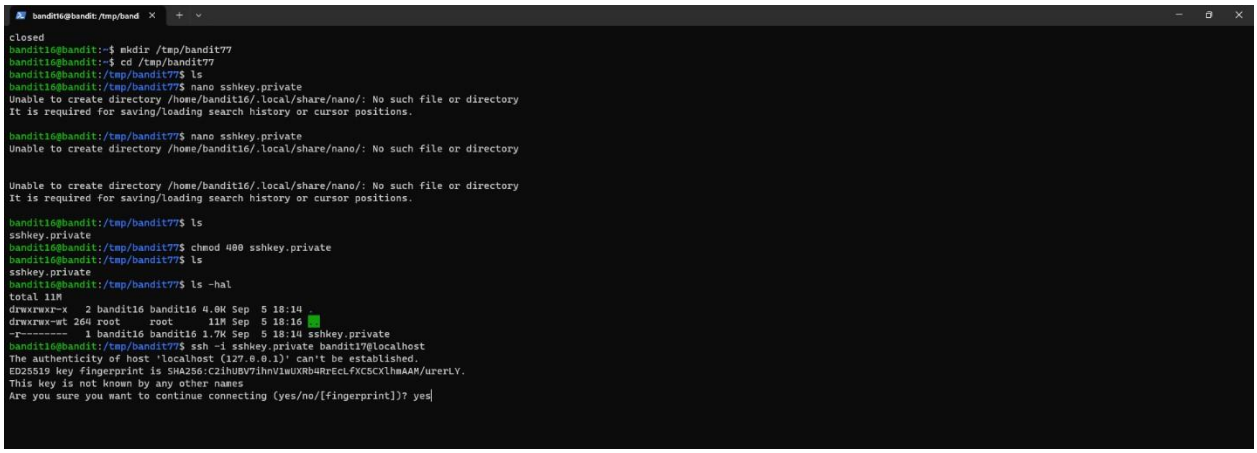
Use “Openssl” and connect to this port on localhost.



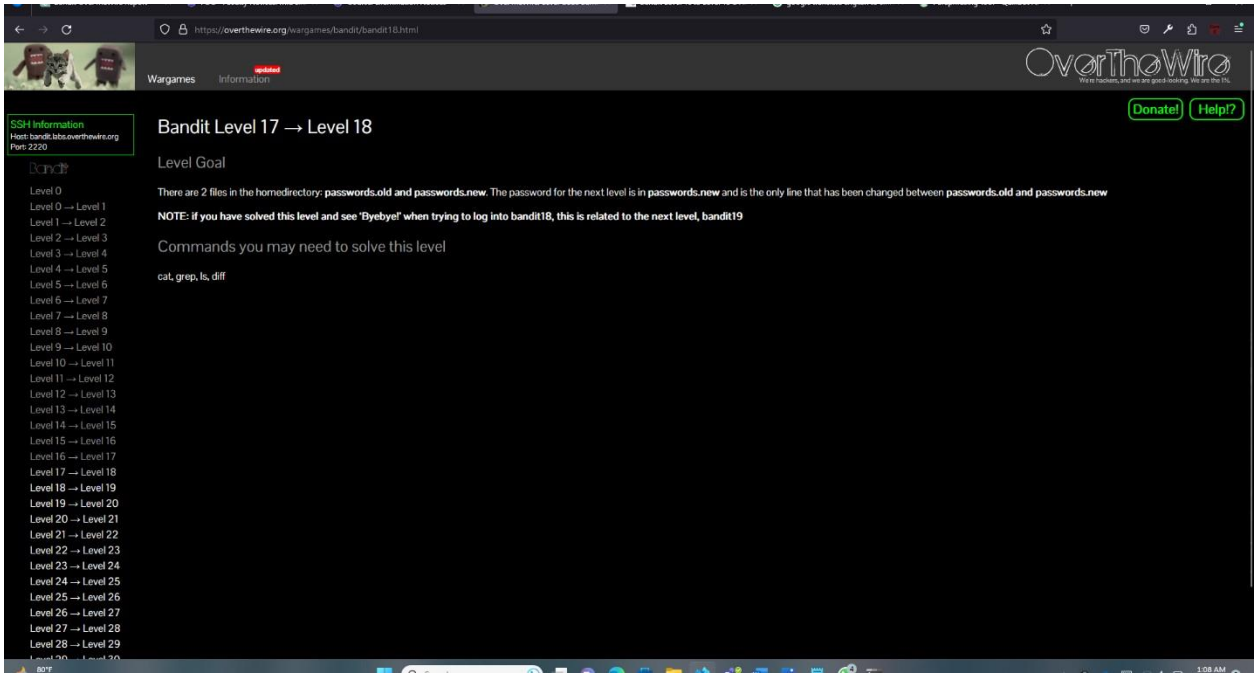
Save this key locally. Use a ssh private key.

“ssh -i sshkey.private bandit17@bandit.labs.overthewire.org -p 2220”

But this file could be edited by the owner and was readable by the group and the entire world. Using a command, we switch the owner’s account to read-only mode. Now can get the access to the Bandit17.

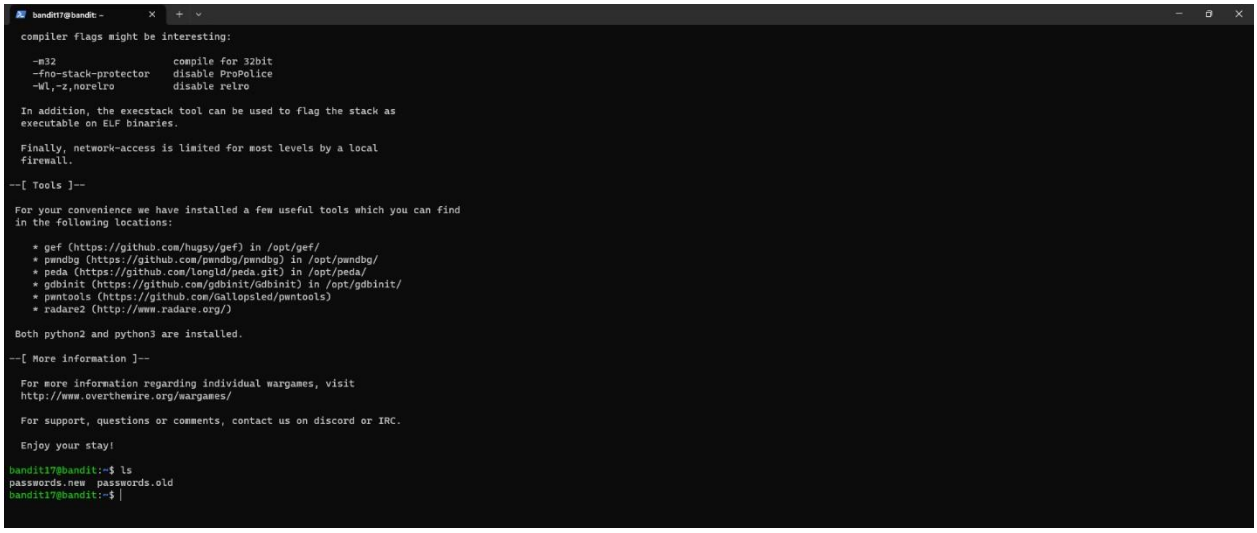


Bandit17 -> Bandit18

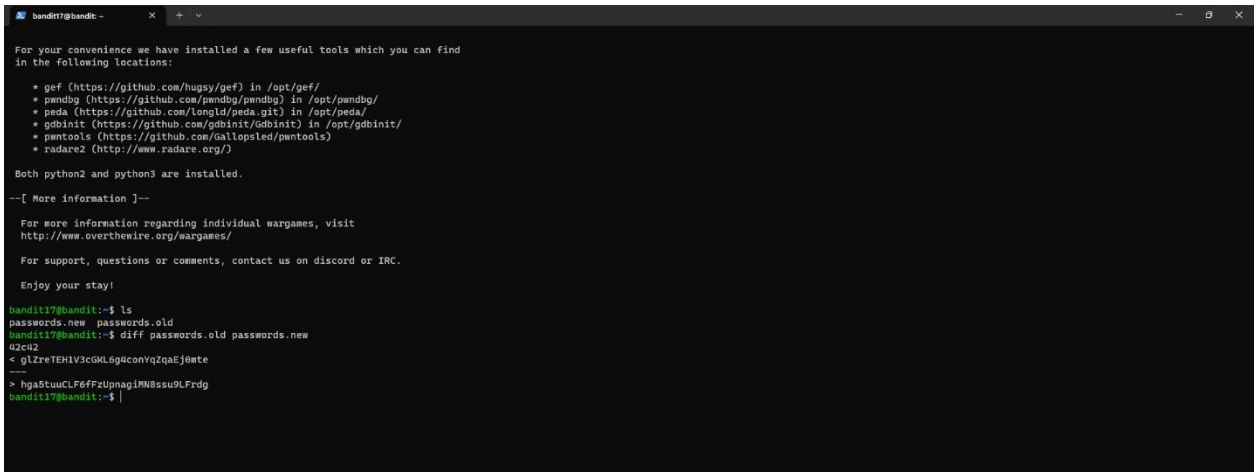


diff – program compares files line by line.

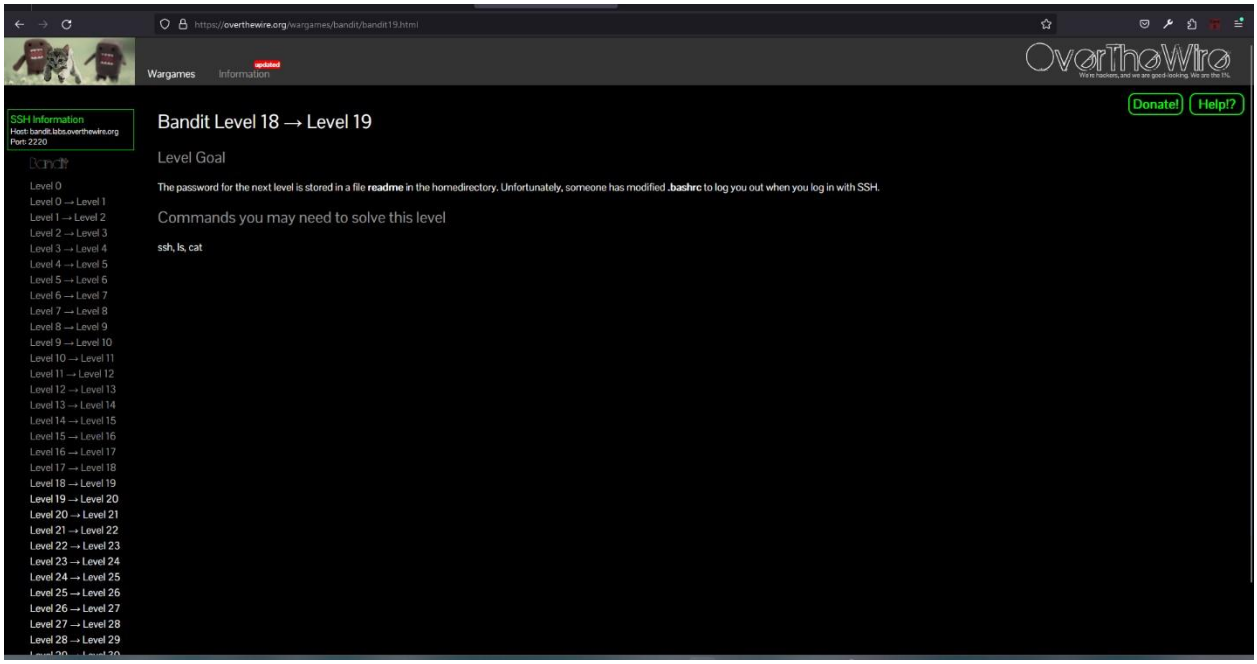
Run the ls command and get the file name.



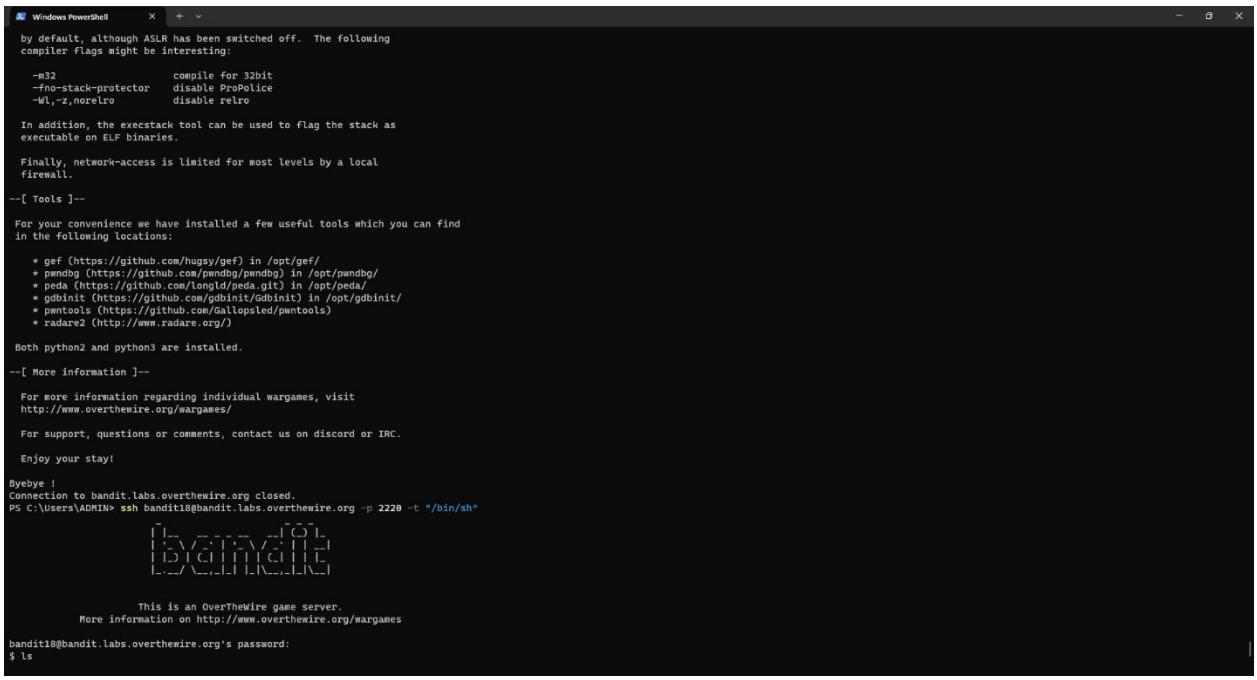
Now run the “diff passwords.old passwords.new” to get different passwords called old and new.



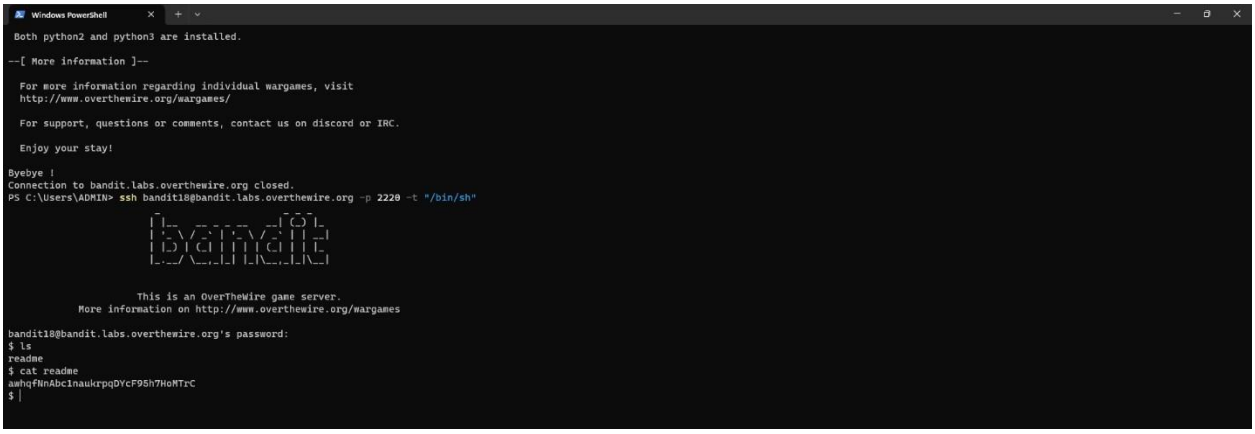
Bandit18 -> Bandit19



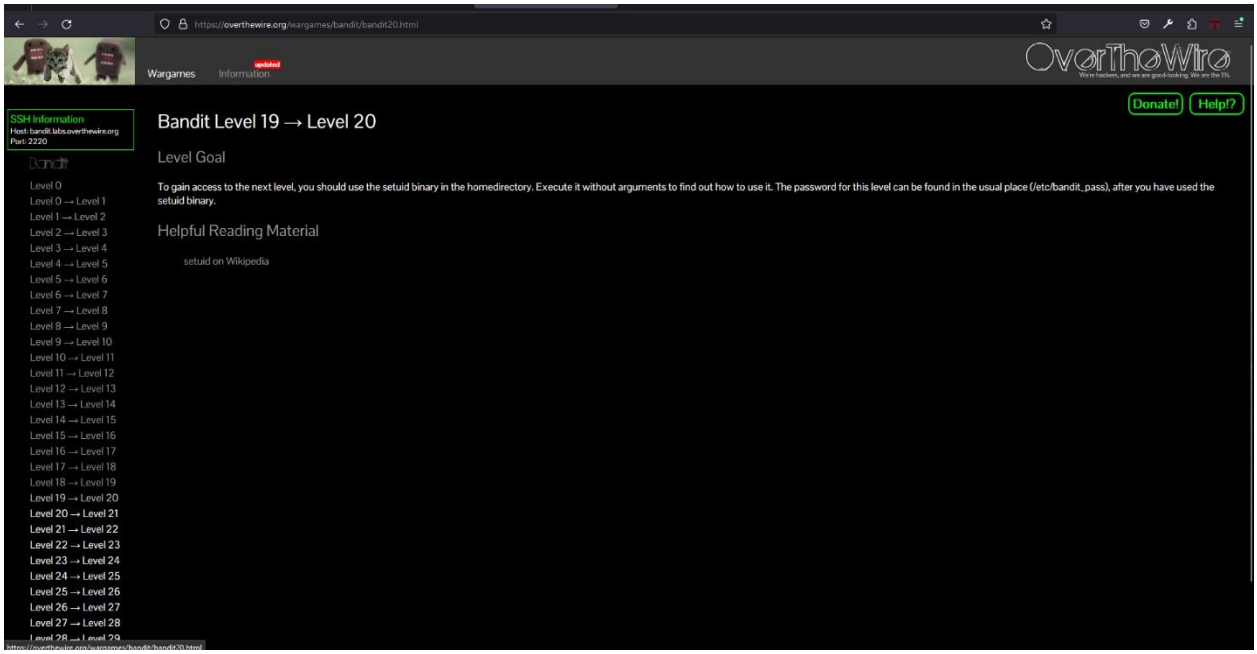
We can try using SSH to log in with them. The terminal window to be used to log into the system is specified using the "-t" flag of the SSH command.



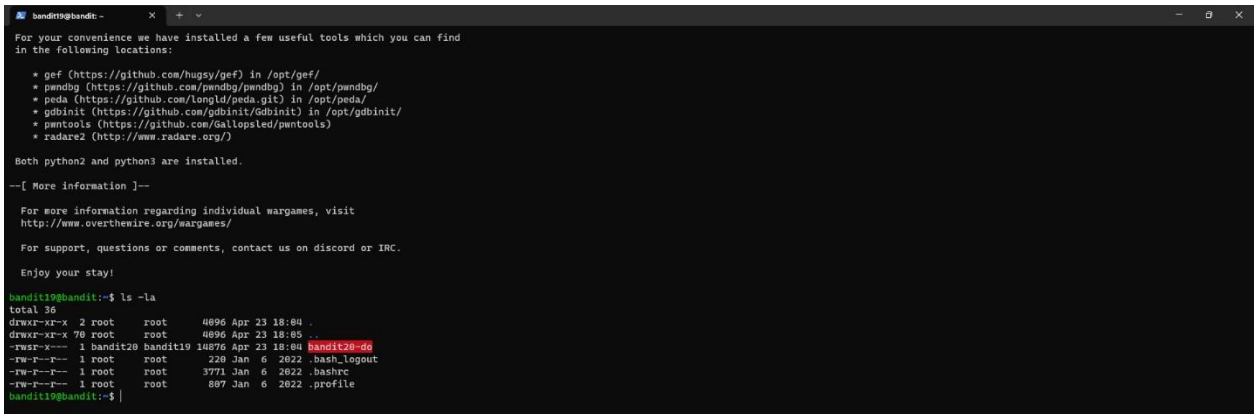
Run the “ls” command and then run the “cat readme” command. When we run those commands wen can find the flag.



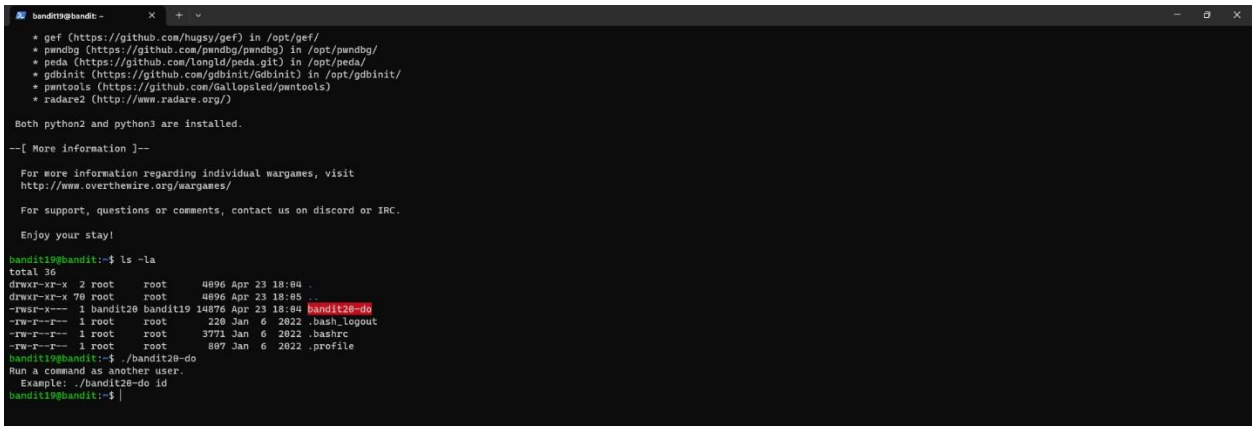
Bandit19 -> Bandit20

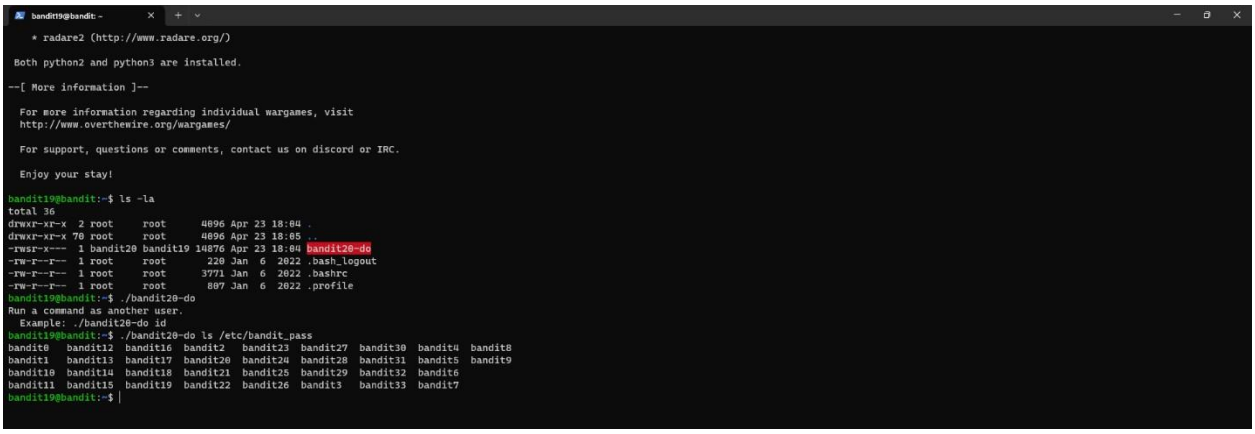


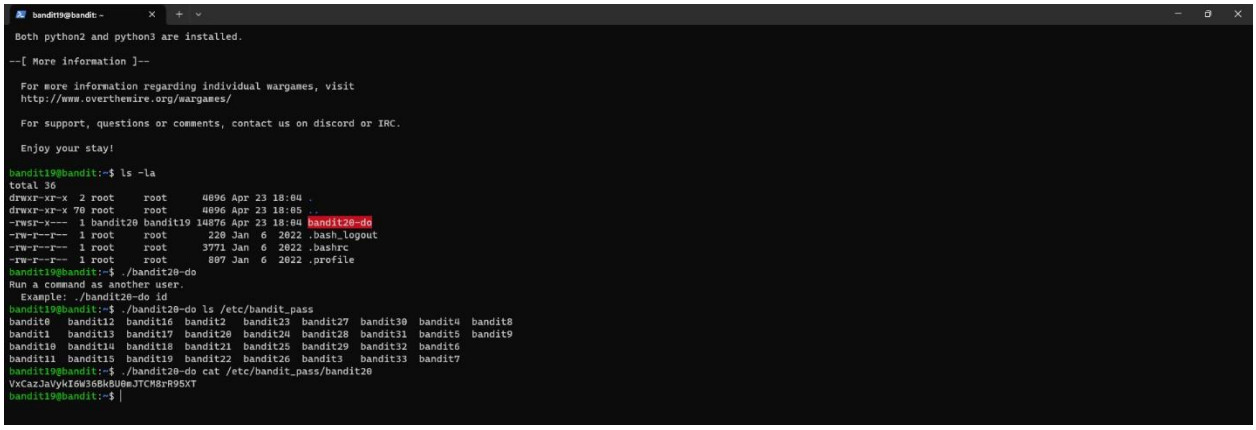
Log into Bandit19 and first we need to check the owner of the setuid binary.



The binary just runs another command as a different user when it is executed, as started. This indicates that we have access to the password file for the Bandit20 user, which is only readable by that user.







Conclusion

I moved to the limits of security as I made my way through the rich pattern of Bandit levels, solving problems that put my skills, creativity, and strength to the test. This investigation has been more than just a practice, it has been a life-changing journey that has expanded my perspectives and strengthened my knowledge of cybersecurity and ethical hacking. In addition to the legal responsibilities that come along with my newfound knowledge, I've learned the value of thorough documentation. My investigation of Bandit levels has expanded not just my knowledge but also my understanding of Linux security. I have not reached the end of my journey in the ethical hacking spirit. It gives an open welcome to everyone who wants to start their own learning and mastering skills missions.

References

1) Medium –
    - https://david-varghese.medium.com/overthewire-bandit-level-16-level-17c137701b3af1
    - https://medium.com/@theGirlWhoEncrypts/overthewire-bandit-level-12-level-13e5b687760d15

2) YouTube –
    - https://www.youtube.com/watch?v=hvSFPyqLizw
    - https://www.youtube.com/watch?v=H8L0P5oKcP0

3) OverTheWire - https://overthewire.org/wargames/bandit/bandit20.html