Useful Unix Commands

Prepared for OverthewireBandit Levels

Prepared by

Bashetty Arun Kumar

Master of Computer Security

22-December-2018

Overthegame (Bandit)

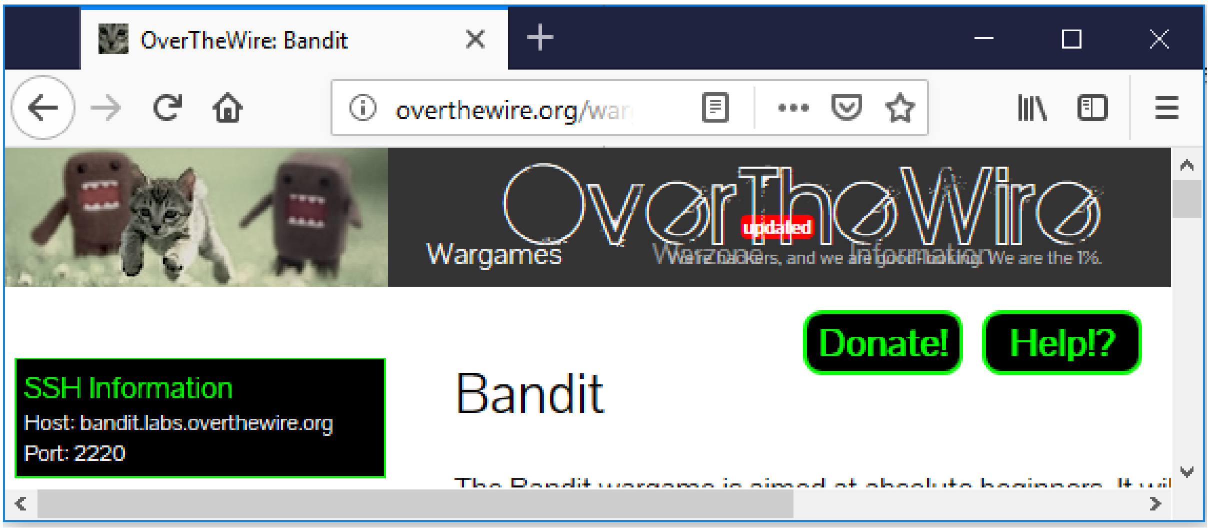
EXECUTIVE SUMMARY

The Bandit wargame is aimed at absolute beginners. It will teach the basics needed to be able to play other wargames.

This game, like most other games, is organised in levels. If you clear the level then only you can step into other Level, otherwise you will not be checked into next Level.

Used website for all LEVELS : http://overthewire.org/wargames/bandit/

Here is the Proof:



SSH Information
Host: bandit.labs.overthewire.org
Port: 2220

Bandit

The Bandit wargame is aimed at absolute beginners. It wil

For clearing all Levels, I used Kali Unix and lot of Commands.

After completing All levels, I found that some levels are more easy and some levels are more depth.

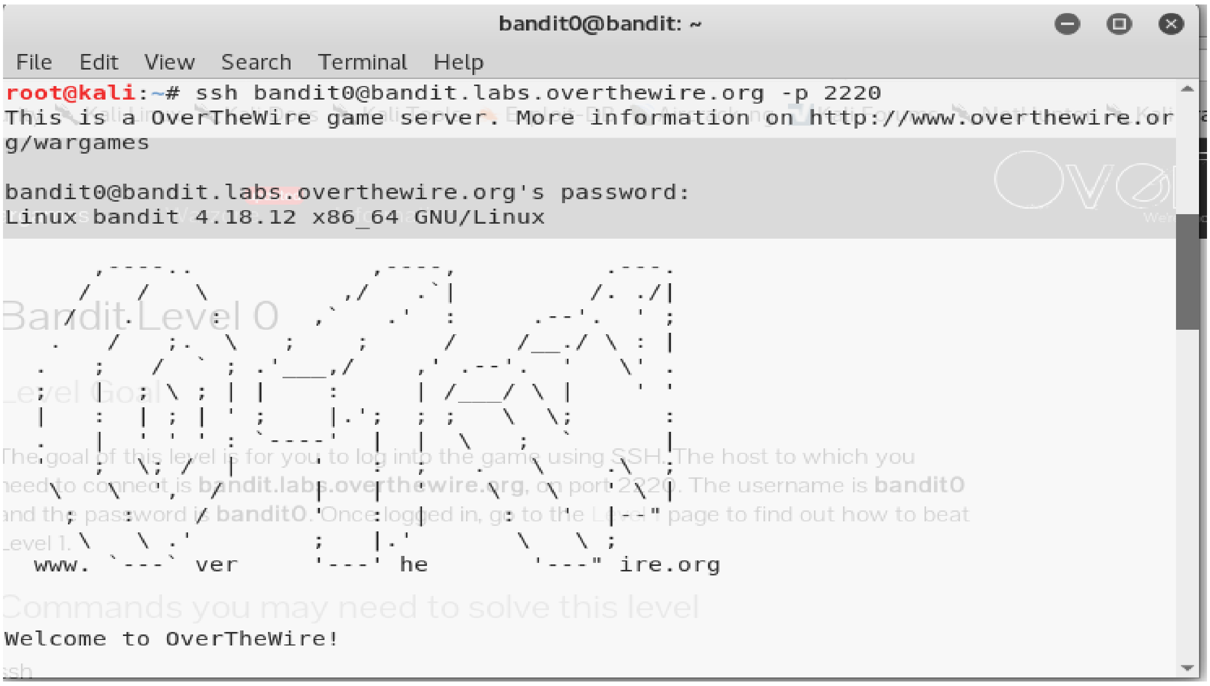
Bandit Level 0

The goal of this level is for you to log into the game using SSH. The host to which you need to connect is **bandit.labs.overthewire.org**, on port 2220. The username is **bandit0** and the password is **bandit0**. Once logged in, go to the Level 1 page to find out how to beat Level 1.

Using Secure Shell (SSH) on Wikipedia, I found that how can I connect using ssh command.

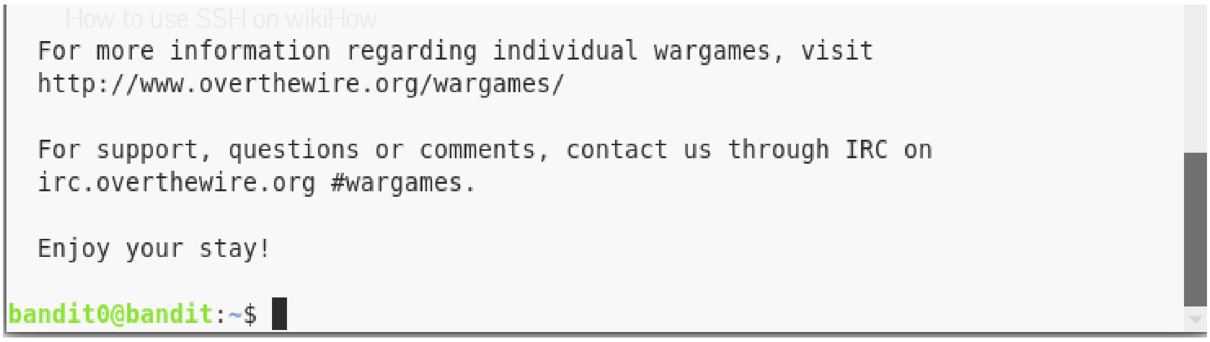
ssh bandit0: bandit.labs.ovethewire.org -p 2220

Password: **bandit0**.



```
root@kali:~# ssh bandit0@bandit.labs.overthewire.org -p 2220
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

bandit0@bandit.labs.overthewire.org's password:
Linux bandit 4.18.12 x86_64 GNU/Linux

      ,----..            ,----,          .---.
     /   /   \         ,/   .`|         /. ./|
    /   .     :      ,`   .'  :     .--'.  ' ;
   .   /   ;.  \   ;    ;     /    /__./ \ : |
  .   ;   /  ` ; .'___,/    ,'.--'.  '   \' .
  ;   |  ; \ ; | |    :     |/__/ \ |    ' '
  |   :  | ; | ' ;    |.';  ;\   \ \;      :
  .   |  ' ' ' : `----'  |  | \   ;  `      |
  '   ;  \; /  |     '   :  ;  .   \    .\  ;
   \   \  ',  /      |   |  '   \   \   ' \ |
    ;   :    /       '   :  |    :   '  |--"
     \   \ .'        ;   |.'      \   \ ;
  www. `---` ver     '---' he      '---" ire.org

Welcome to OverTheWire!
```

Now , you can see user is chaged from root to bandit0,



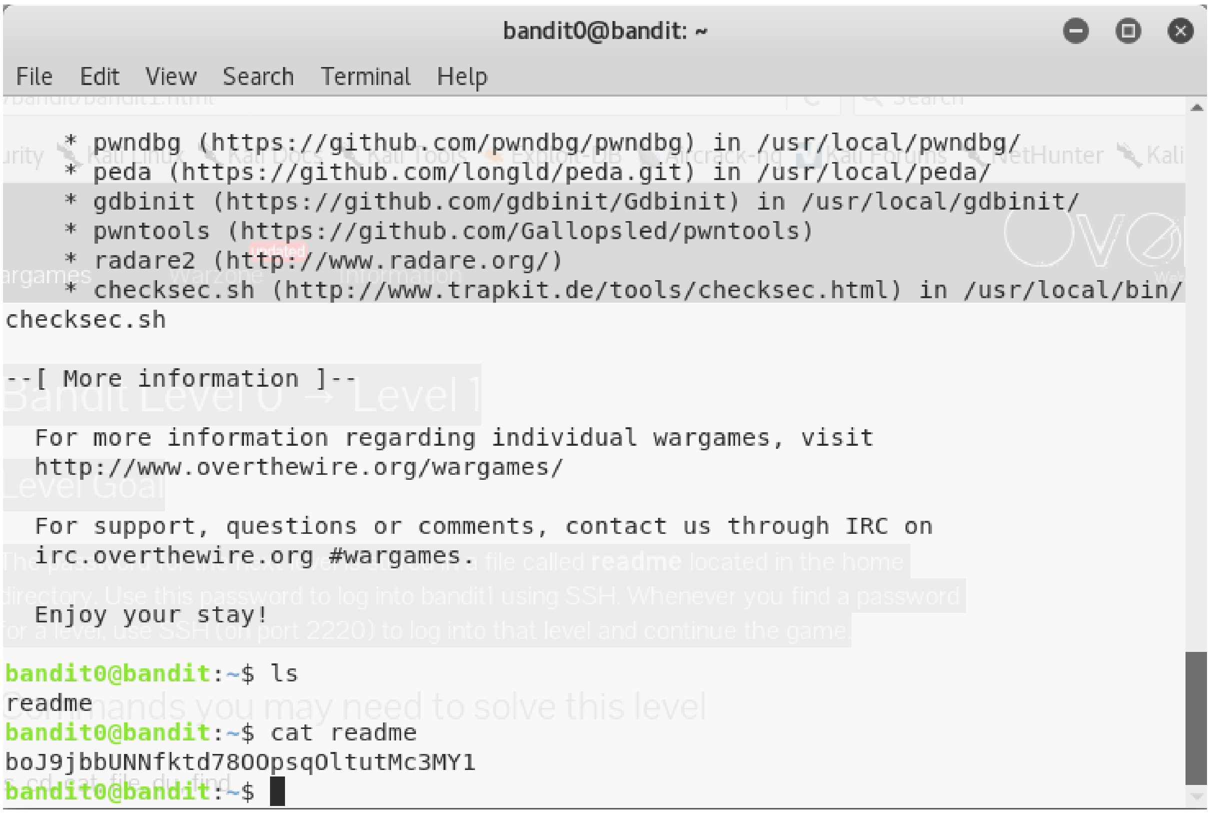
```
For more information regarding individual wargames, visit
http://www.overthewire.org/wargames/

For support, questions or comments, contact us through IRC on
irc.overthewire.org #wargames.

Enjoy your stay!

bandit0@bandit:~$ 
```

Bandit Level 0 → Level 1

Level Goal

The password for the next level is stored in a file called **readme** located in the home directory. Use this password to log into bandit1 using SSH. Whenever you find a password for a level, use SSH (on port 2220) to log into that level and continue the game.

Log out from bandit0 user by using "**exit**" command and every time you can use the ssh command to login into next user.



```
    * pwndbg (https://github.com/pwndbg/pwndbg) in /usr/local/pwndbg/
    * peda (https://github.com/longld/peda.git) in /usr/local/peda/
    * gdbinit (https://github.com/gdbinit/Gdbinit) in /usr/local/gdbinit/
    * pwntools (https://github.com/Gallopsled/pwntools)
    * radare2 (http://www.radare.org/)
    * checksec.sh (http://www.trapkit.de/tools/checksec.html) in /usr/local/bin/
checksec.sh

--[ More information ]--

  For more information regarding individual wargames, visit
  http://www.overthewire.org/wargames/

  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit0@bandit:~$ ls
readme
bandit0@bandit:~$ cat readme
boJ9jbbUNNfktd78OOpsqOltutMc3MY1
bandit0@bandit:~$ 
```

Using above all commands you will get a password inside readme file,

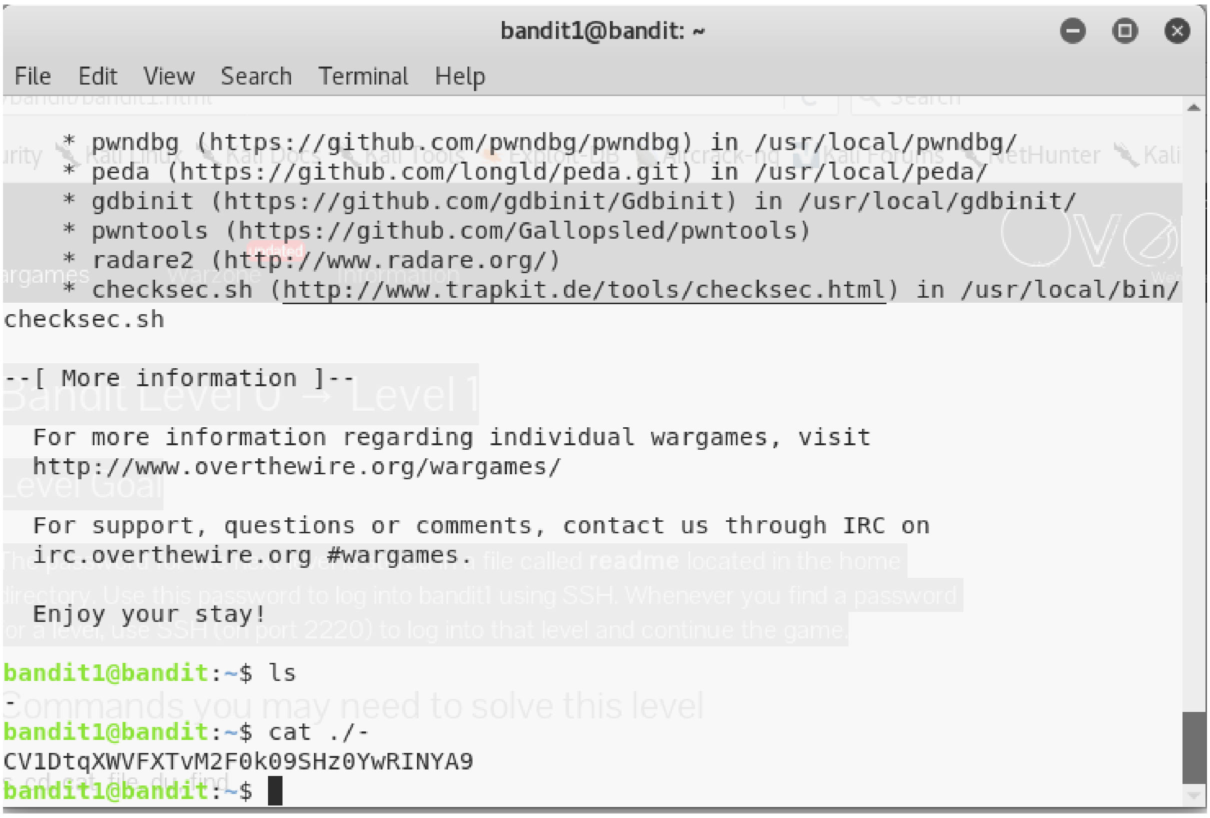
Password for next Level is:

boJ9jbbUNNfktd78OOpsqOltutMc3MY1

Bandit Level 1 → Level 2

Level Goal

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



```
    * pwndbg (https://github.com/pwndbg/pwndbg) in /usr/local/pwndbg/
    * peda (https://github.com/longld/peda.git) in /usr/local/peda/
    * gdbinit (https://github.com/gdbinit/Gdbinit) in /usr/local/gdbinit/
    * pwntools (https://github.com/Gallopsled/pwntools)
    * radare2 (http://www.radare.org/)
    * checksec.sh (http://www.trapkit.de/tools/checksec.html) in /usr/local/bin/
checksec.sh

--[ More information ]--

  For more information regarding individual wargames, visit
  http://www.overthewire.org/wargames/

  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit1@bandit:~$ ls
-
bandit1@bandit:~$ cat ./-
CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9
bandit1@bandit:~$ 
```

For hidden files, you can use the command cat ./-, it will display all hidden files inside the

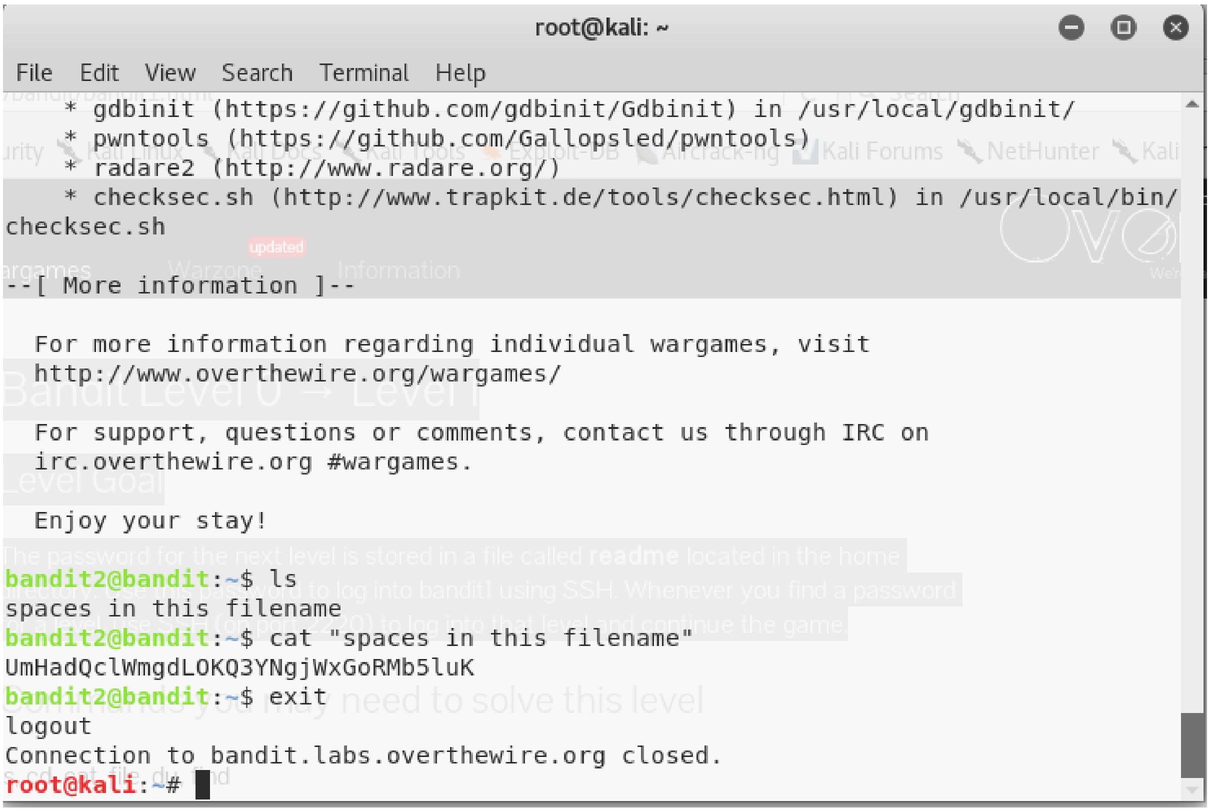
Password for next Level is:

CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9

Bandit Level 2 → Level 3

Level Goal

The password for the next level is stored in a file called **spaces in this filename** located in the home directory.



```
   * gdbinit (https://github.com/gdbinit/Gdbinit) in /usr/local/gdbinit/
   * pwntools (https://github.com/Gallopsled/pwntools)
   * radare2 (http://www.radare.org/)
   * checksec.sh (http://www.trapkit.de/tools/checksec.html) in /usr/local/bin/
checksec.sh

--[ More information ]--

  For more information regarding individual wargames, visit
  http://www.overthewire.org/wargames/

  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit2@bandit:~$ ls
spaces in this filename
bandit2@bandit:~$ cat "spaces in this filename"
UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK
bandit2@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

You can See above using exit command, you can exist from bandit2 level and its automatically changed to root as a user.

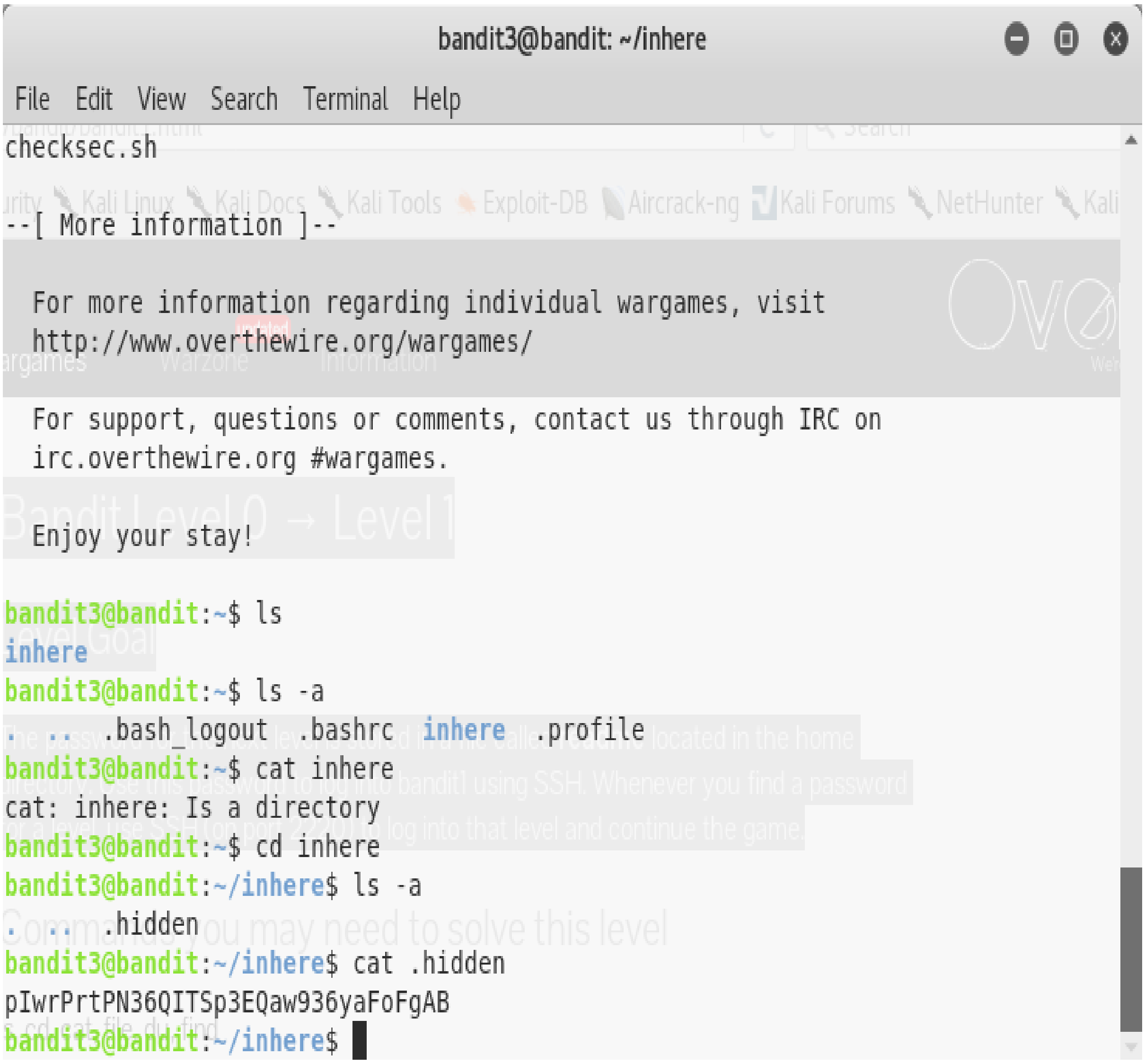
Password for next Level is:

UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK

Bandit Level 3 → Level 4

Level Goal

The password for the next level is stored in a hidden file in the **inhere** directory.



```
checksec.sh

--[ More information ]--

  For more information regarding individual wargames, visit
  http://www.overthewire.org/wargames/

  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit3@bandit:~$ ls
inhere
bandit3@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  inhere  .profile
bandit3@bandit:~$ cat inhere
cat: inhere: Is a directory
bandit3@bandit:~$ cd inhere
bandit3@bandit:~/inhere$ ls -a
.  ..  .hidden
bandit3@bandit:~/inhere$ cat .hidden
pIwrPrtPN36QITSp3EQaw936yaFoFgAB
bandit3@bandit:~/inhere$
```

Using ls -a, you can see the hidden file inside the directory

Using cat command, you can display the output for next level.

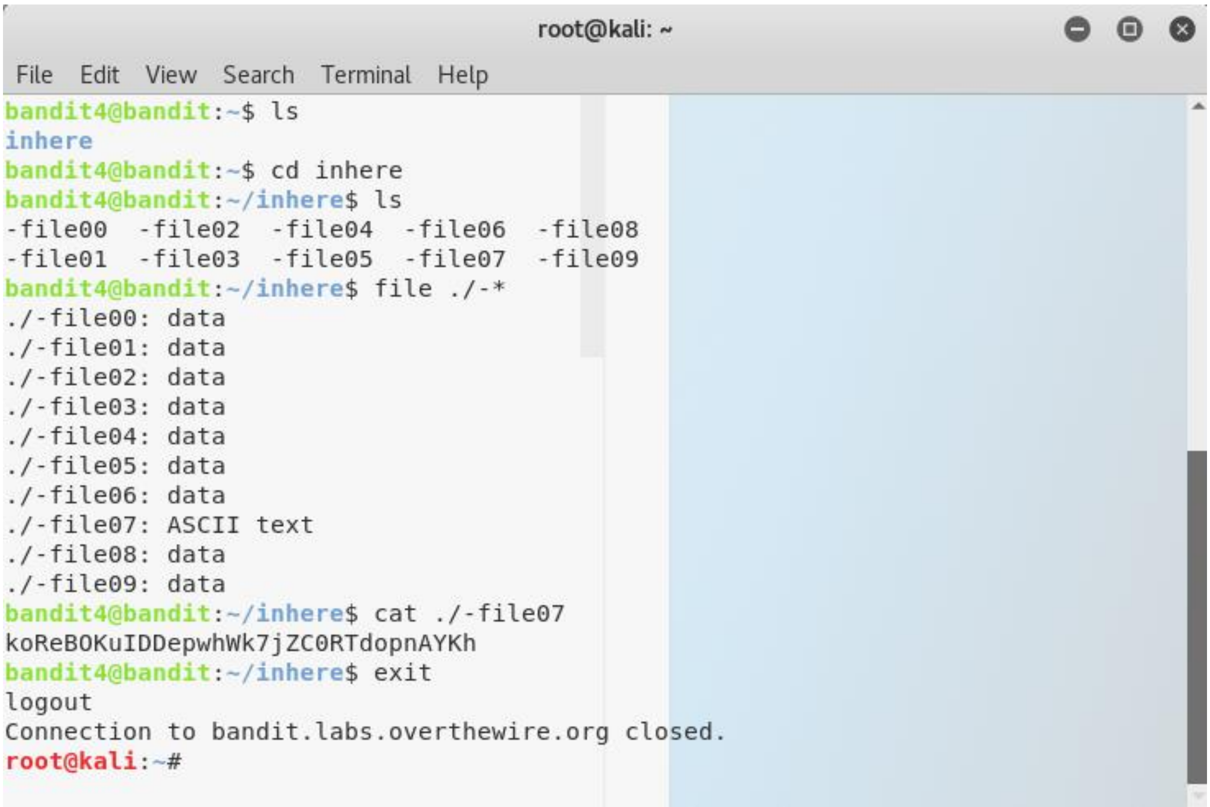
Password for next Level is:

pIwrPrtPN36QITSp3EQaw936yaFoFgAB

Bandit Level 4 → Level 5

Level Goal

The password for the next level is stored in the only human-readable file in the **inhere** directory. Tip: if your terminal is messed up, try the “reset” command.



```
bandit4@bandit:~$ ls
inhere
bandit4@bandit:~$ cd inhere
bandit4@bandit:~/inhere$ ls
-file00  -file02  -file04  -file06  -file08
-file01  -file03  -file05  -file07  -file09
bandit4@bandit:~/inhere$ file ./-*
./-file00: data
./-file01: data
./-file02: data
./-file03: data
./-file04: data
./-file05: data
./-file06: data
./-file07: ASCII text
./-file08: data
./-file09: data
bandit4@bandit:~/inhere$ cat ./-file07
koReBOKuIDDepwhWk7jZC0RTdopnAYKh
bandit4@bandit:~/inhere$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

I am using ls command and there is “inhere” file,

Using **file./-*** command, I can get a file type, there file07 is ASCII text

Using command, it will display the password, Password for the next level is:

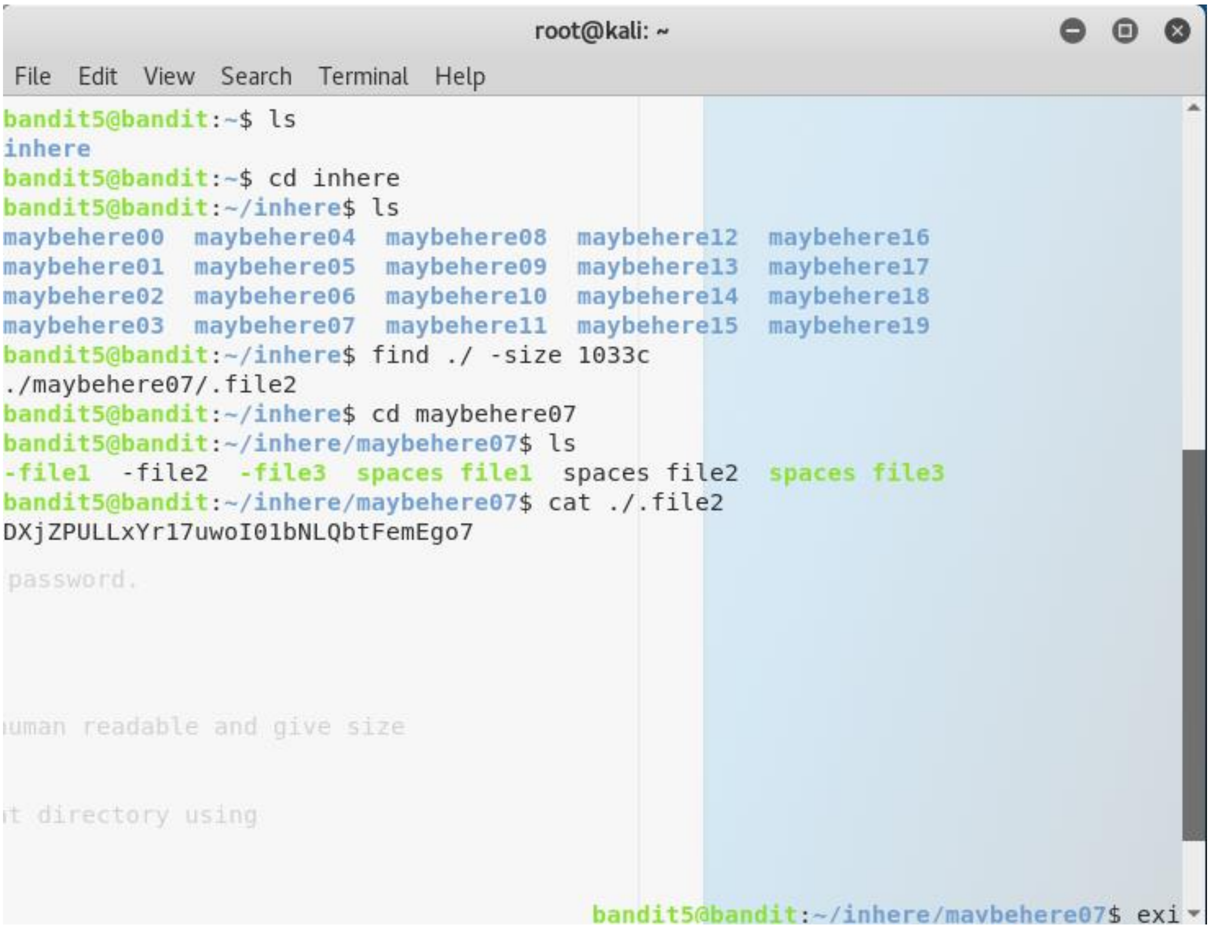
koReBOKuIDDepwhWk7jZC0RTdopnAYKh

Bandit Level 5 → Level 6

Level Goal

The password for the next level is stored in a file somewhere under the **inhere** directory and has all the following properties:

- human-readable
- 1033 bytes in size
- not executable



```
bandit5@bandit:~$ ls
inhere
bandit5@bandit:~$ cd inhere
bandit5@bandit:~/inhere$ ls
maybehere00  maybehere04  maybehere08  maybehere12  maybehere16
maybehere01  maybehere05  maybehere09  maybehere13  maybehere17
maybehere02  maybehere06  maybehere10  maybehere14  maybehere18
maybehere03  maybehere07  maybehere11  maybehere15  maybehere19
bandit5@bandit:~/inhere$ find ./ -size 1033c
./maybehere07/.file2
bandit5@bandit:~/inhere$ cd maybehere07
bandit5@bandit:~/inhere/maybehere07$ ls
-file1  -file2  -file3  spaces file1  spaces file2  spaces file3
bandit5@bandit:~/inhere/maybehere07$ cat ./.file2
DXjZPULLxYr17uwoI01bNLQbtFemEgo7

password.



uman readable and give size


t directory using


                                        bandit5@bandit:~/inhere/maybehere07$ exi
```

In hint, it is showing the human readable code means use command find./ to see all files, in 2nd, shows gave size and use command -size 1033c

The total command is find ./ -size 1033c to password it is showing inside

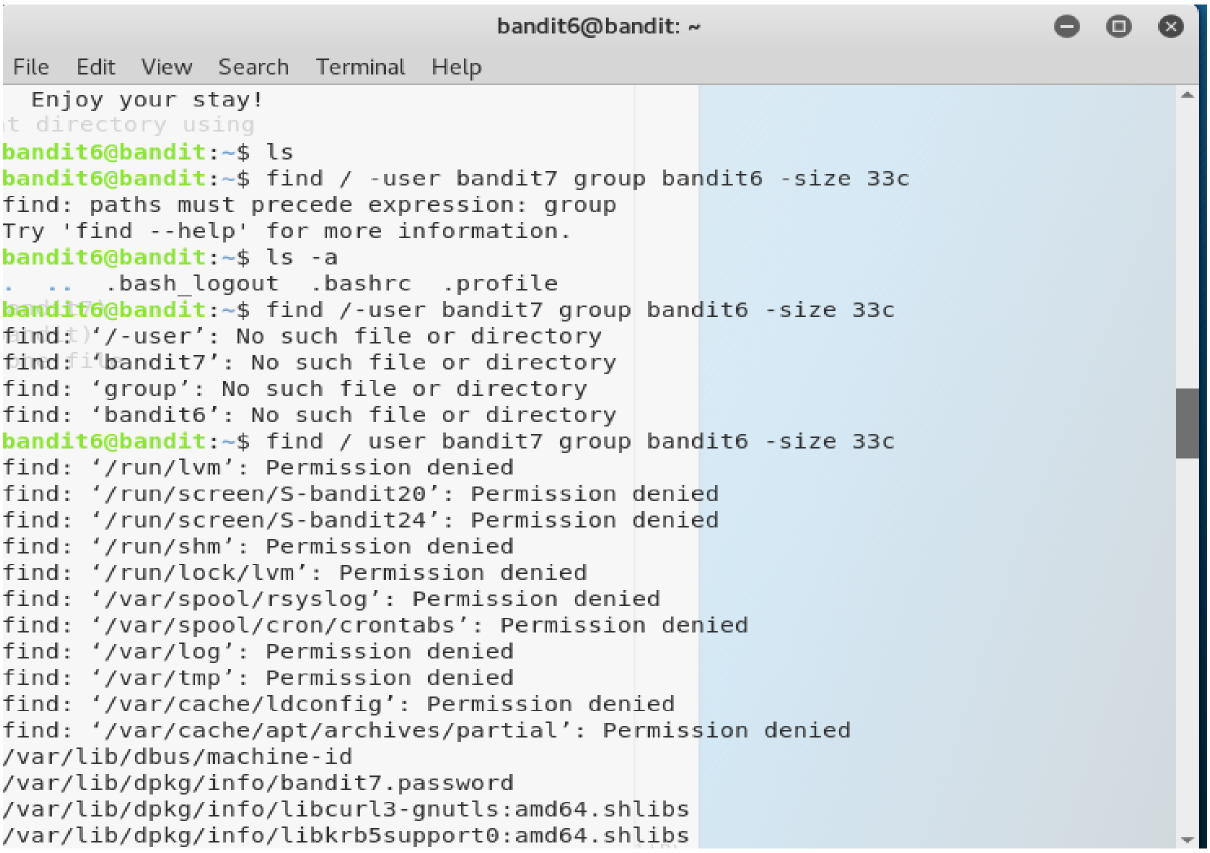
The Password is: **DXjZPULLxYr17uwoI01bNLQbtFemEgo7**

Bandit Level 6 → Level 7

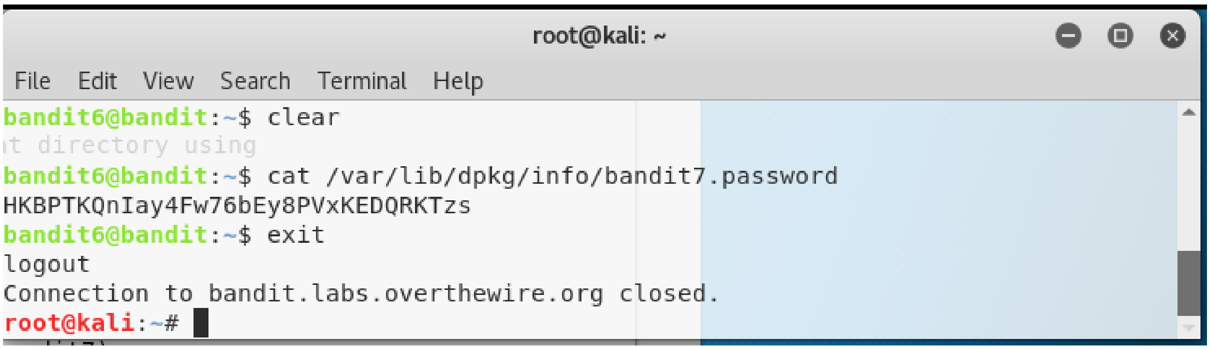
Level Goal

The password for the next level is stored **somewhere on the server** and has all of the following properties:

- owned by user bandit7
- owned by group bandit6
- 33 bytes in size



```
 Enjoy your stay!
t directory using
bandit6@bandit:~$ ls
bandit6@bandit:~$ find / -user bandit7 group bandit6 -size 33c
find: paths must precede expression: group
Try 'find --help' for more information.
bandit6@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  .profile
bandit6@bandit:~$ find /-user bandit7 group bandit6 -size 33c
find: ‘/-user’: No such file or directory
find: ‘bandit7’: No such file or directory
find: ‘group’: No such file or directory
find: ‘bandit6’: No such file or directory
bandit6@bandit:~$ find / user bandit7 group bandit6 -size 33c
find: ‘/run/lvm’: Permission denied
find: ‘/run/screen/S-bandit20’: Permission denied
find: ‘/run/screen/S-bandit24’: Permission denied
find: ‘/run/shm’: Permission denied
find: ‘/run/lock/lvm’: Permission denied
find: ‘/var/spool/rsyslog’: Permission denied
find: ‘/var/spool/cron/crontabs’: Permission denied
find: ‘/var/log’: Permission denied
find: ‘/var/tmp’: Permission denied
find: ‘/var/cache/ldconfig’: Permission denied
find: ‘/var/cache/apt/archives/partial’: Permission denied
/var/lib/dbus/machine-id
/var/lib/dpkg/info/bandit7.password
/var/lib/dpkg/info/libcurl3-gnutls:amd64.shlibs
/var/lib/dpkg/info/libkrb5support0:amd64.shlibs
```



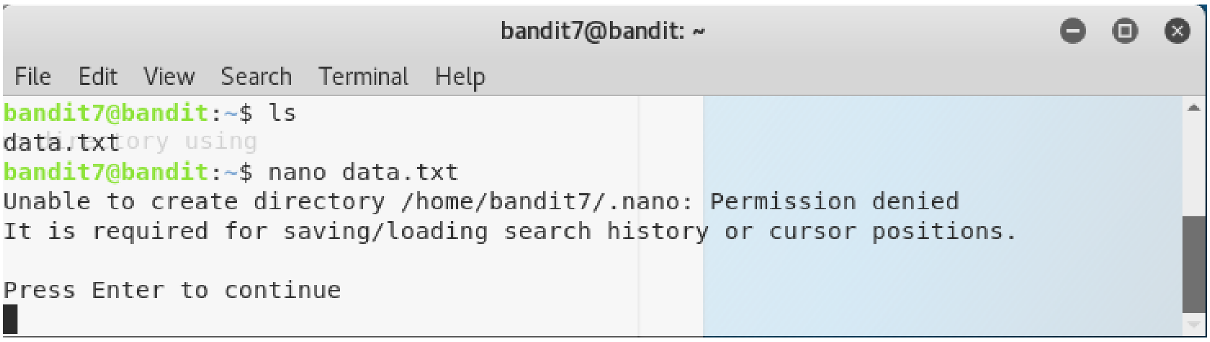
```
bandit6@bandit:~$ clear
t directory using
bandit6@bandit:~$ cat /var/lib/dpkg/info/bandit7.password
HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs
bandit6@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~# 
```

Password for the next Level is: **HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs**

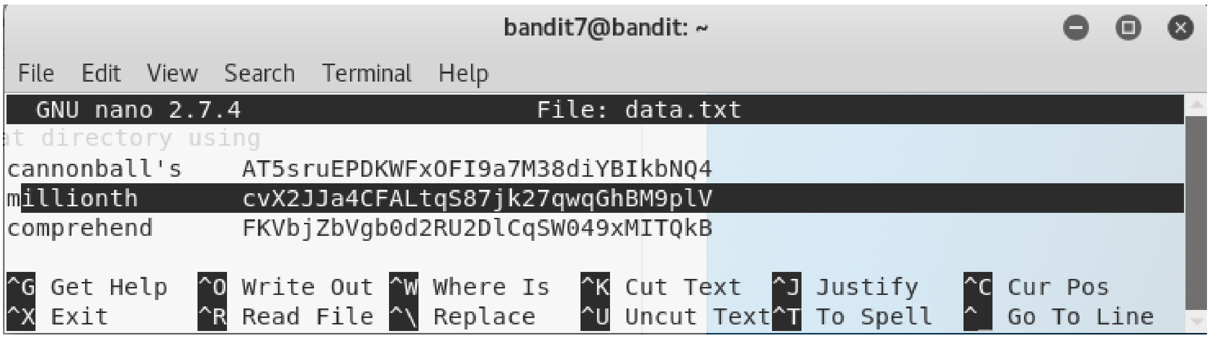
Bandit Level 7 → Level 8

Level Goal

The password for the next level is stored in the file **data.txt** next to the word **millionth**



```
bandit7@bandit:~$ ls
data.txt
bandit7@bandit:~$ nano data.txt
Unable to create directory /home/bandit7/.nano: Permission denied
It is required for saving/loading search history or cursor positions.

Press Enter to continue
```



```
  GNU nano 2.7.4                        File: data.txt

cannonball's    AT5sruEPDKWFxOFI9a7M38diYBIkbNQ4
millionth       cvX2JJa4CFALtqS87jk27qwqGhBM9plV
comprehend      FKVbjZbVgb0d2RU2DlCqSW049xMITQkB

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Spell  ^_ Go To Line
```

Using nano command we can see the data inside the file and you can change, if you have permissions,

In above we opened the file using nano command, in hint it will show word millionth,

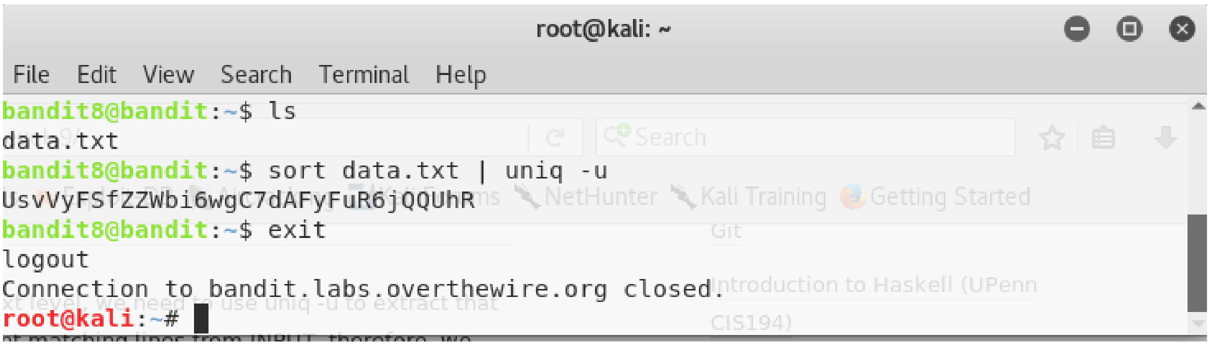
So, I searched the word using CNTRL+R and type millionth,

The password for the next level is: **cvX2JJa4CFALtqS87jk27qwqGhBM9plV**

Bandit Level 8 → Level 9

Level Goal

The password for the next level is stored in the file **data.txt** and is the only line of text that occurs only once



```
bandit8@bandit:~$ ls
data.txt
bandit8@bandit:~$ sort data.txt | uniq -u
UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR
bandit8@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~# █
```

For this level, we can use the basic command is sort to sorting the elements inside the data.txt

Sort:

Using uniq -u command, we can pick only unique code inside the data.txt file

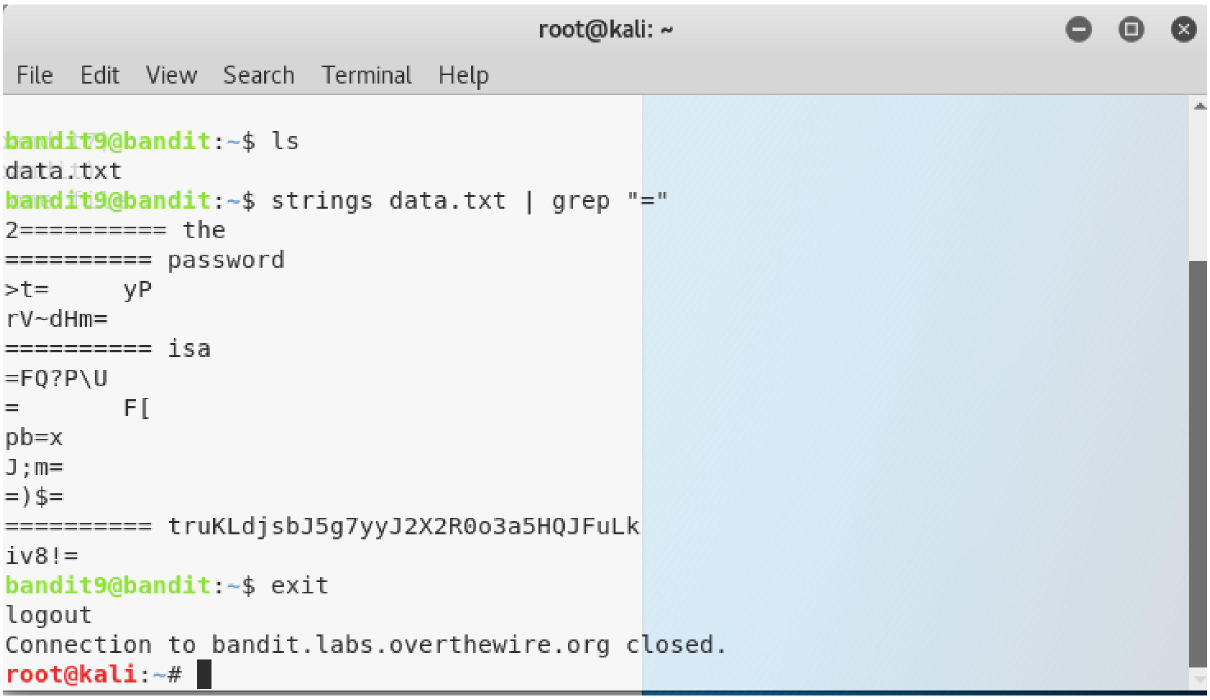
Password for the next Level is:

UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR

Bandit Level 9 → Level 10

Level Goal

The password for the next level is stored in the file **data.txt** in one of the few human-readable strings, beginning with several '=' characters.



```
bandit9@bandit:~$ ls
data.txt
bandit9@bandit:~$ strings data.txt | grep "="
2========== the
========== password
>t=     yP
rV~dHm=
========== isa
=FQ?P\U
=       F[
pb=x
J;m=
=)$=
========== truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk
iv8!=
bandit9@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

Using simple and important command using grep, we can solve this level,

The Strings command basically prints the strings of printable characters in file (data.txt)

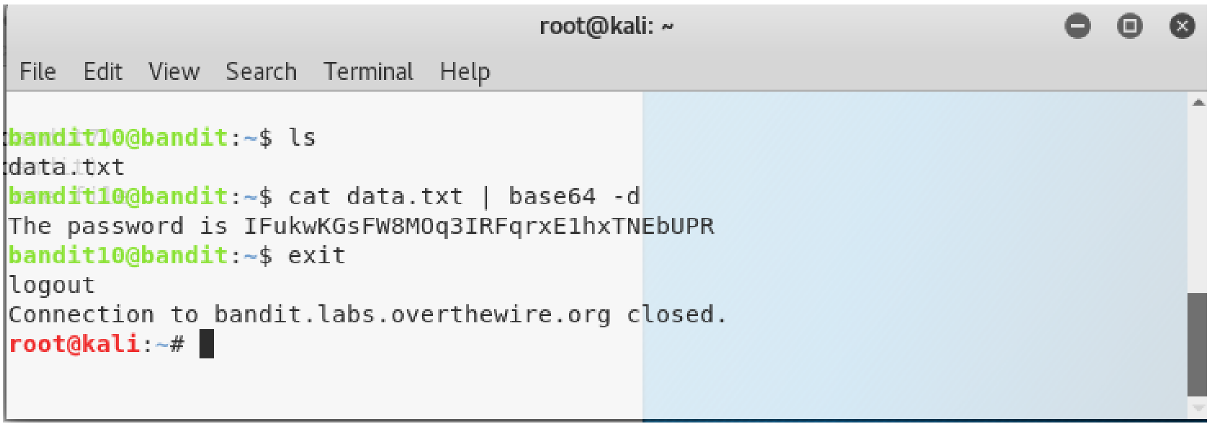
The password for the next Level is:

truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk

Bandit Level 10 → Level 11

Level Goal

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



```
bandit10@bandit:~$ ls
data.txt
bandit10@bandit:~$ cat data.txt | base64 -d
The password is IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR
bandit10@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

In this level **Base64** -d command is very important.

Using ls , command you can see the list inside the file and directories

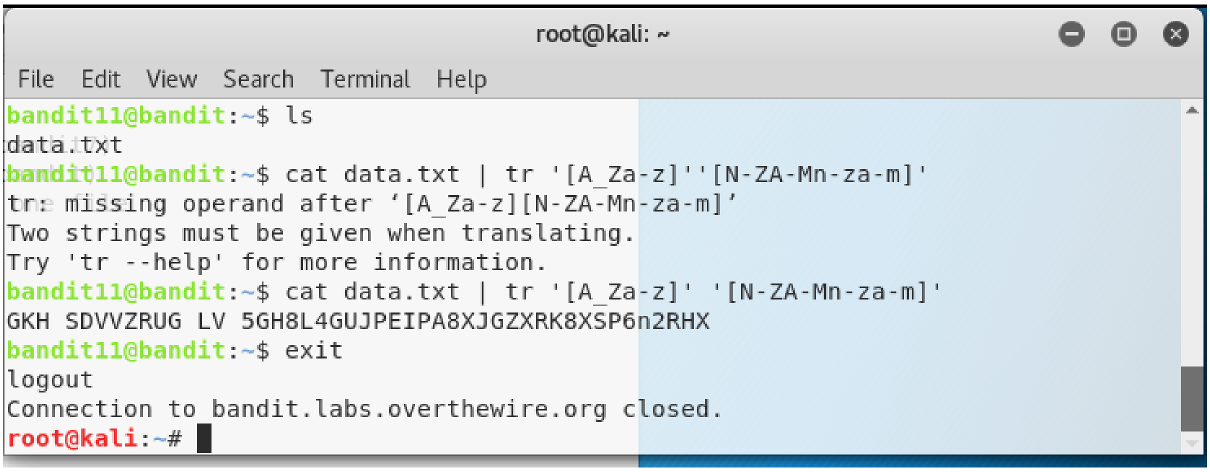
Base64 is a group of similar binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation.

The password the next level: **IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR**

Bandit Level 11 → Level 12

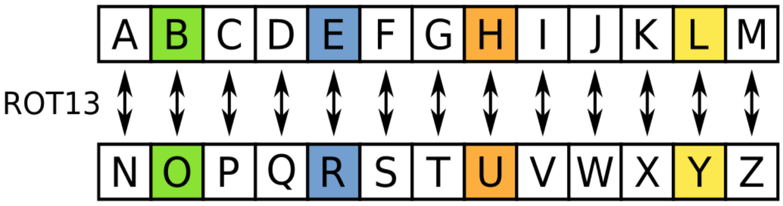
Level Goal

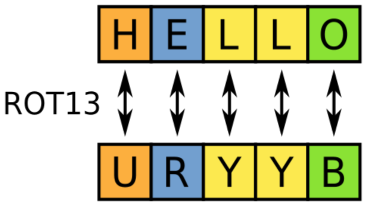
The password for the next level is stored in the file **data.txt**, where all lowercase (a-z) and uppercase (A-Z) letters have been rotated by 13 positions



```
bandit11@bandit:~$ ls
data.txt
bandit11@bandit:~$ cat data.txt | tr '[A_Za-z]''[N-ZA-Mn-za-m]'
tr: missing operand after ‘[A_Za-z][N-ZA-Mn-za-m]’
Two strings must be given when translating.
Try 'tr --help' for more information.
bandit11@bandit:~$ cat data.txt | tr '[A_Za-z]' '[N-ZA-Mn-za-m]'
GKH SDVVZRUG LV 5GH8L4GUJPEIPA8XJGZXRK8XSP6n2RHX
bandit11@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~# 
```

In this level, you can use the helpful material forRot13**, ROT13** replaces each letter by its partner 13 characters further along the alphabet.





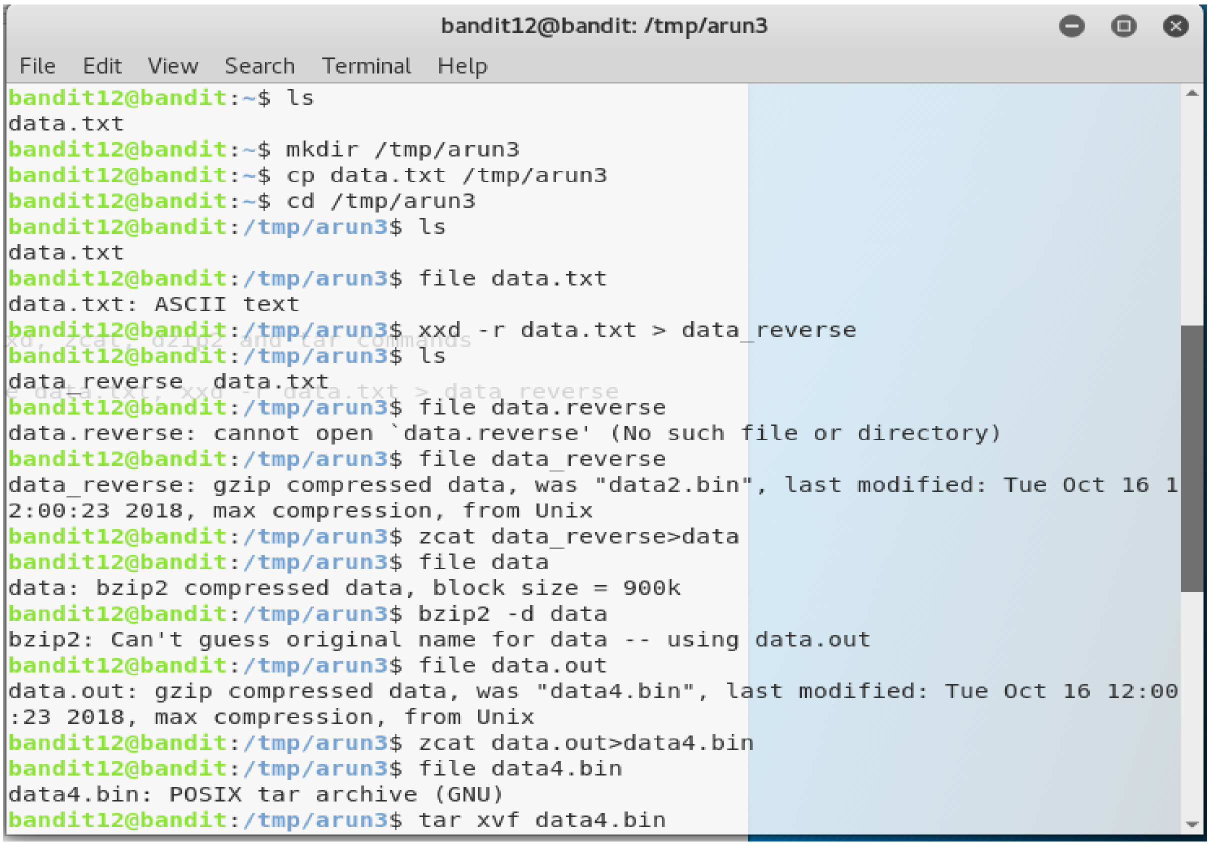
The password for the next Level is:

5Te8Y4drgCRfCx8ugdwuEX8KFC6k2EUu

Bandit Level 12 → Level 13

Level Goal

The password for the next level is stored in the file **data.txt**, which is a hexdump of a file that has been repeatedly compressed. For this level it may be useful to create a directory under /tmp in which you can work using mkdir. For example: mkdir /tmp/myname123. Then copy the datafile using cp, and rename it using mv (read the manpages)



```
bandit12@bandit:~$ ls
data.txt
bandit12@bandit:~$ mkdir /tmp/arun3
bandit12@bandit:~$ cp data.txt /tmp/arun3
bandit12@bandit:~$ cd /tmp/arun3
bandit12@bandit:/tmp/arun3$ ls
data.txt
bandit12@bandit:/tmp/arun3$ file data.txt
data.txt: ASCII text
bandit12@bandit:/tmp/arun3$ xxd -r data.txt > data_reverse
bandit12@bandit:/tmp/arun3$ ls
data_reverse  data.txt
bandit12@bandit:/tmp/arun3$ file data.reverse
data.reverse: cannot open `data.reverse' (No such file or directory)
bandit12@bandit:/tmp/arun3$ file data_reverse
data_reverse: gzip compressed data, was "data2.bin", last modified: Tue Oct 16 1
2:00:23 2018, max compression, from Unix
bandit12@bandit:/tmp/arun3$ zcat data_reverse>data
bandit12@bandit:/tmp/arun3$ file data
data: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/arun3$ bzip2 -d data
bzip2: Can't guess original name for data -- using data.out
bandit12@bandit:/tmp/arun3$ file data.out
data.out: gzip compressed data, was "data4.bin", last modified: Tue Oct 16 12:00
:23 2018, max compression, from Unix
bandit12@bandit:/tmp/arun3$ zcat data.out>data4.bin
bandit12@bandit:/tmp/arun3$ file data4.bin
data4.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/arun3$ tar xvf data4.bin
```

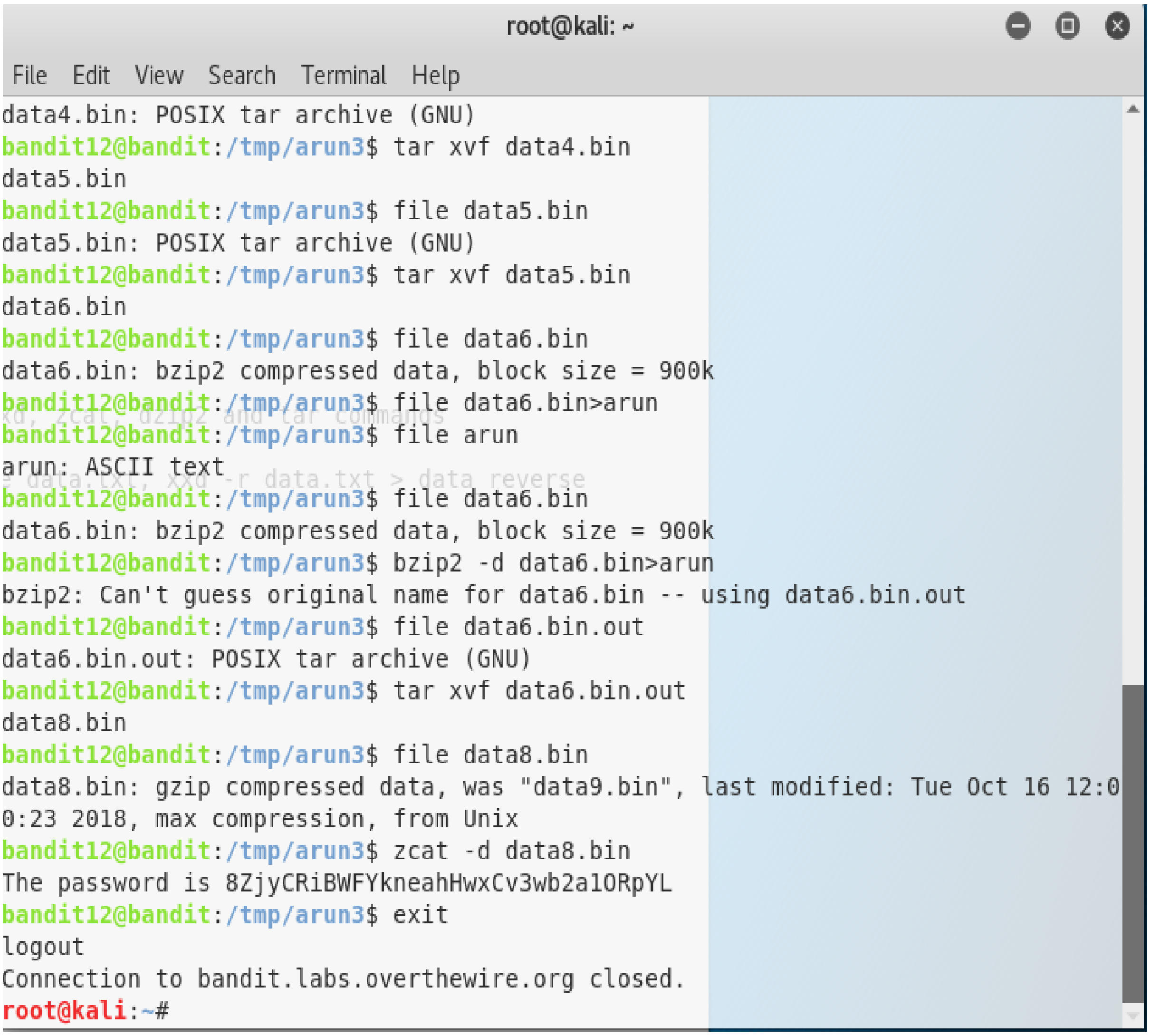
For this Level, we need to know about, Xxd, zcat, bzip2 and tar xvf commands

Xxd -r filename > new filename (reverse the hexdump),

Zcat filename>new filename (used to gzip compressed data)

Bzip2 -d filename (used to bzip2 compressed data)

As it is showing in hint that make a new directory inside the /tmp folder. Because the data.txt file is hexdump file and we compress lot of time, so I Created a directory inside the **/tmp/arun3.**



```
data4.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/arun3$ tar xvf data4.bin
data5.bin
bandit12@bandit:/tmp/arun3$ file data5.bin
data5.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/arun3$ tar xvf data5.bin
data6.bin
bandit12@bandit:/tmp/arun3$ file data6.bin
data6.bin: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/arun3$ file data6.bin>arun
bandit12@bandit:/tmp/arun3$ file arun
arun: ASCII text
bandit12@bandit:/tmp/arun3$ file data6.bin
data6.bin: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/arun3$ bzip2 -d data6.bin>arun
bzip2: Can't guess original name for data6.bin -- using data6.bin.out
bandit12@bandit:/tmp/arun3$ file data6.bin.out
data6.bin.out: POSIX tar archive (GNU)
bandit12@bandit:/tmp/arun3$ tar xvf data6.bin.out
data8.bin
bandit12@bandit:/tmp/arun3$ file data8.bin
data8.bin: gzip compressed data, was "data9.bin", last modified: Tue Oct 16 12:0
0:23 2018, max compression, from Unix
bandit12@bandit:/tmp/arun3$ zcat -d data8.bin
The password is 8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL
bandit12@bandit:/tmp/arun3$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

tar xvf filename is used to change the tar archive (GNU) file.

You have to repeat all 3 commands until, you will get a ASCII code.

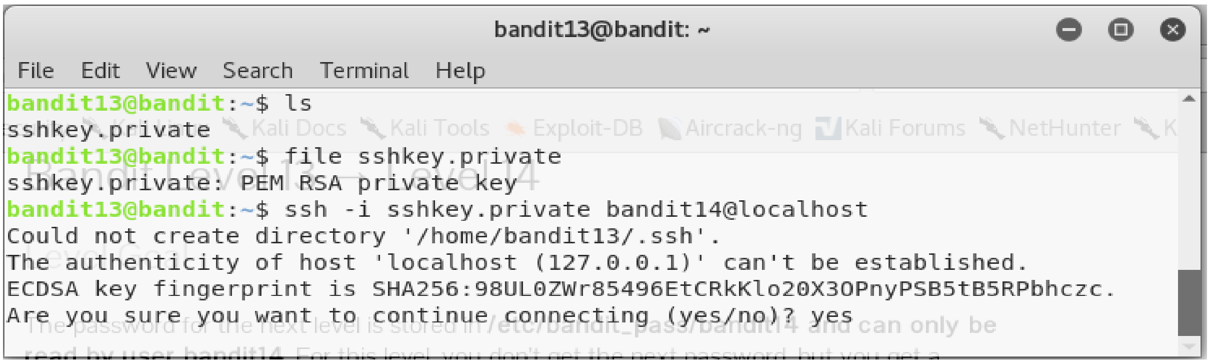
it will finally file showing ASCII password for the next Level is:

8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL

Bandit Level 13 → Level 14

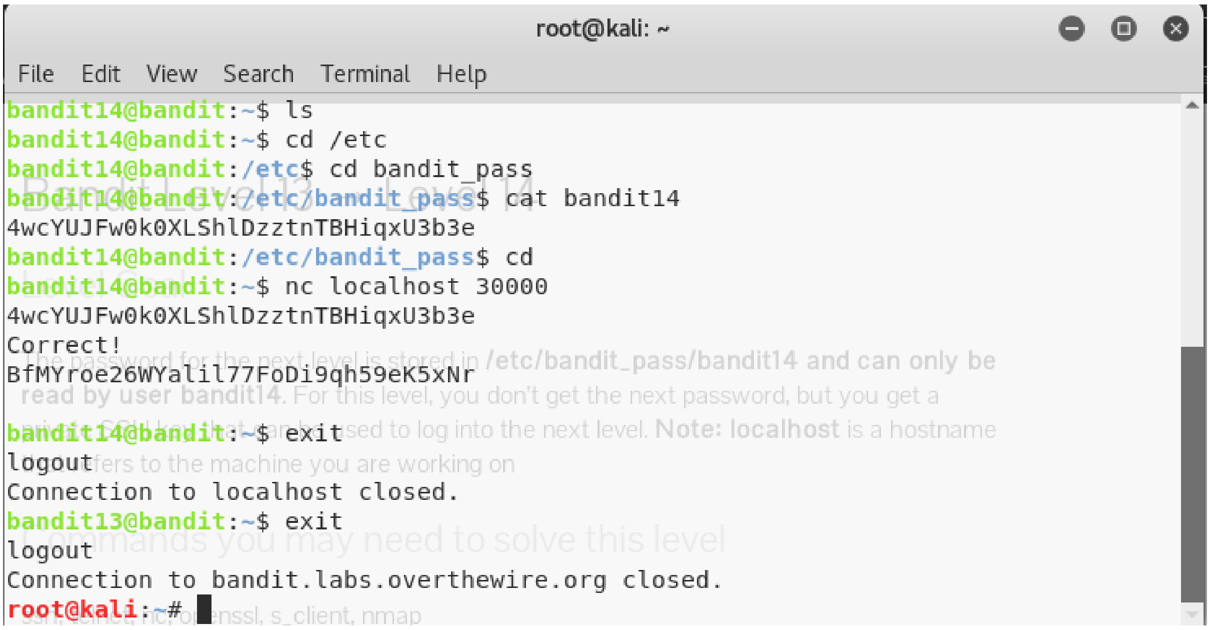
Level Goal

The password for the next level is stored in **/etc/bandit_pass/bandit14 and can only be read by user bandit14**. For this level, you don’t get the next password, but you get a private SSH key that can be used to log into the next level. **Note: localhost** is a hostname that refers to the machine you are working on.



```
bandit13@bandit:~$ ls
sshkey.private
bandit13@bandit:~$ file sshkey.private
sshkey.private: PEM RSA private key
bandit13@bandit:~$ ssh -i sshkey.private bandit14@localhost
Could not create directory '/home/bandit13/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
```

Inside the bandit13, you can see the sshkey.private and i.e., RSA Private key. In this step we will not get password. You can change the username by using localhost is hostname, If I click on yes, it will change to user bandit15



```
bandit14@bandit:~$ ls
bandit14@bandit:~$ cd /etc
bandit14@bandit:/etc$ cd bandit_pass
bandit14@bandit:/etc/bandit_pass$ cat bandit14
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
bandit14@bandit:/etc/bandit_pass$ cd
bandit14@bandit:~$ nc localhost 30000
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
Correct!
BfMYroe26WYalil77FoDi9qh59eK5xNr

bandit14@bandit:~$ exit
logout
Connection to localhost closed.
bandit13@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

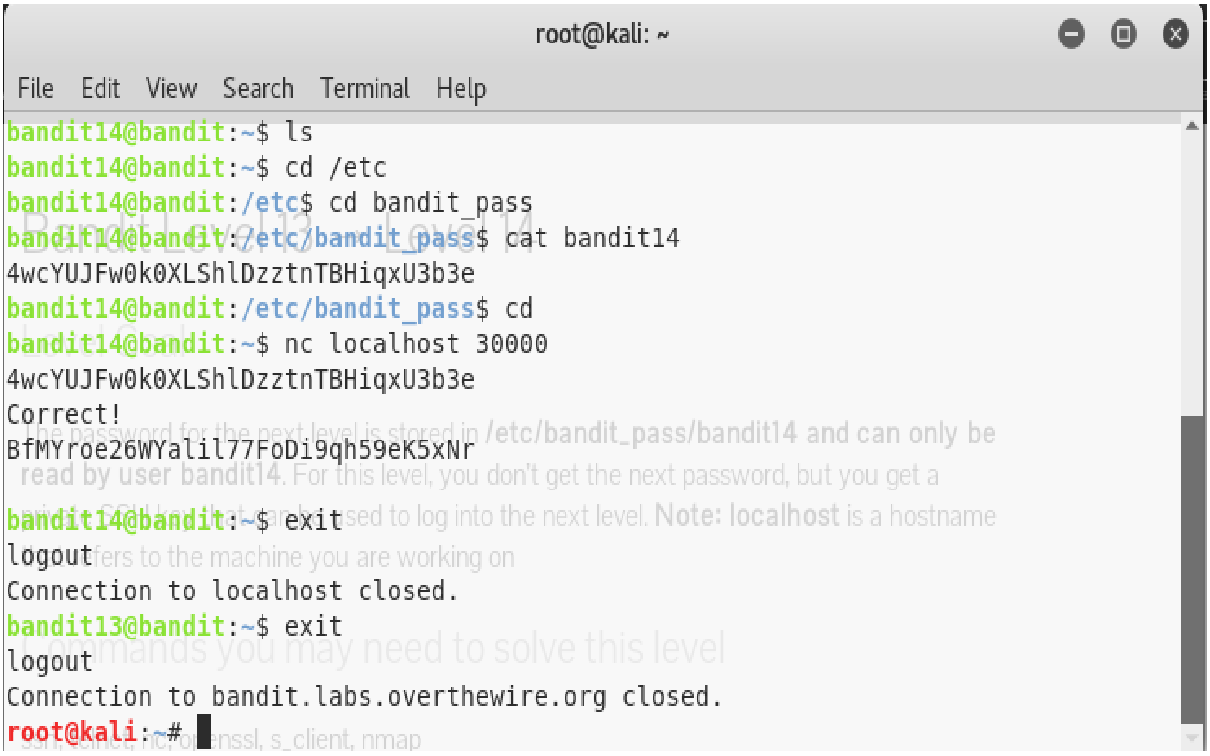
In this above question, it is showing that the password will be inside the /etc/bandit_pass/bandit14.

The password is: **4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e**

Bandit Level 14 → Level 15

Level Goal:

The password for the next level can be retrieved by submitting the password of the current level to **port 30000 on localhost**



```
bandit14@bandit:~$ ls
bandit14@bandit:~$ cd /etc
bandit14@bandit:/etc$ cd bandit_pass
bandit14@bandit:/etc/bandit_pass$ cat bandit14
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
bandit14@bandit:/etc/bandit_pass$ cd
bandit14@bandit:~$ nc localhost 30000
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
Correct!
BfMYroe26WYalil77FoDi9qh59eK5xNr

bandit14@bandit:~$ exit
logout
Connection to localhost closed.
bandit13@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

In this level, you can use the command “nc”, nc referred to NetCat,

The nc (or netcat) utility is used for just about anything under the sun involving TCP or UDP.

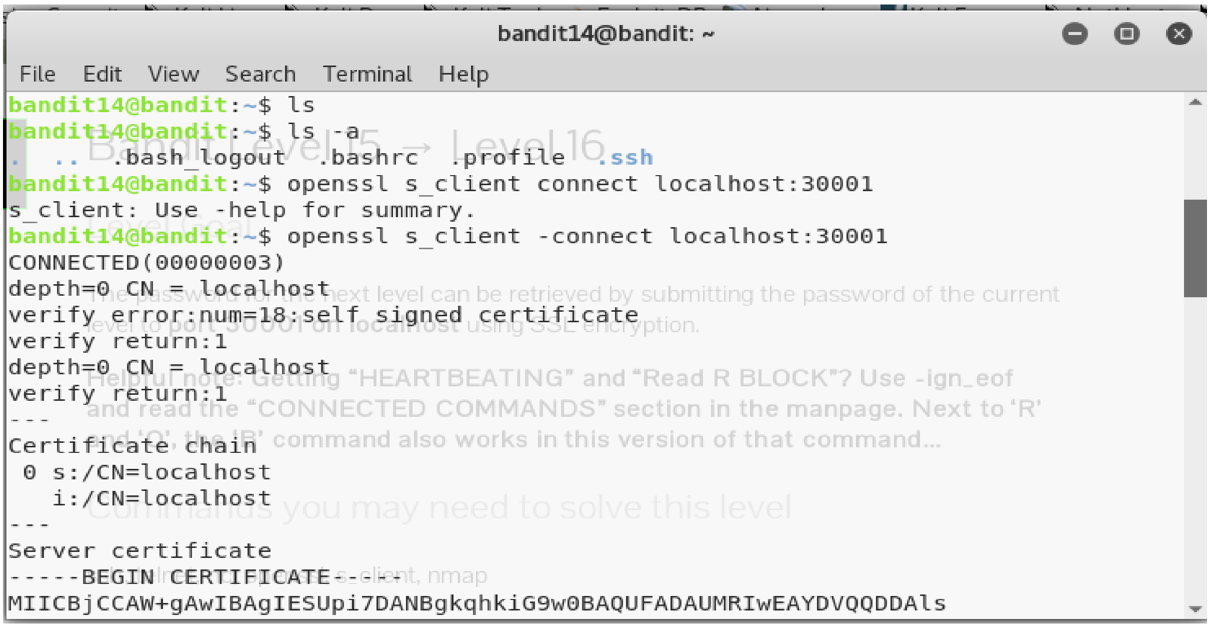
Use the nc command with localhost 30000,

The password is: ***BfMYroe26WYalil77FoDi9qh59eK5xNr***

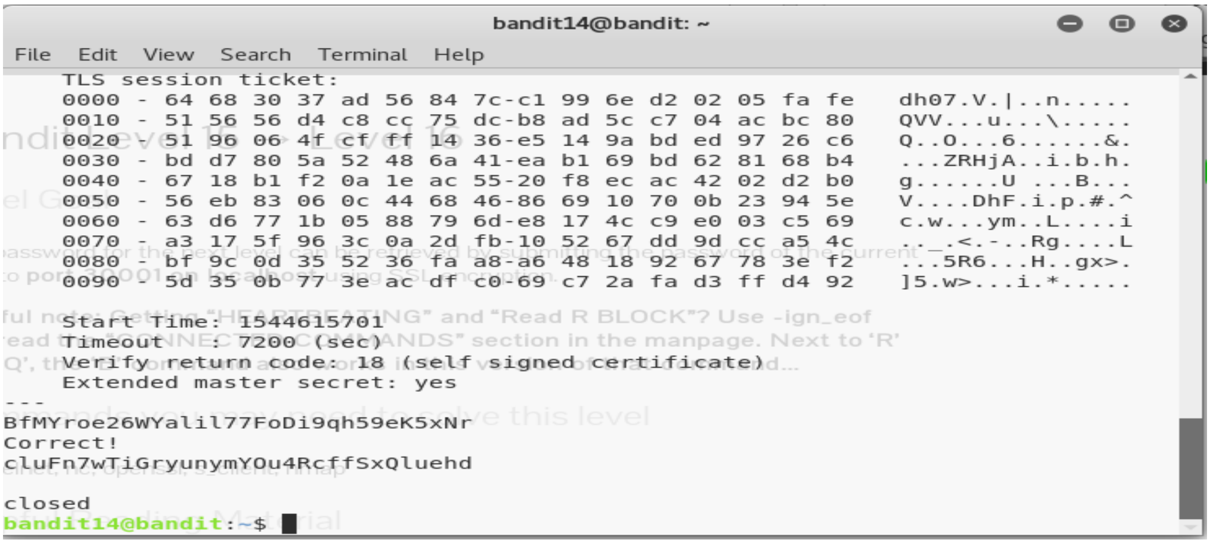
Bandit Level 15 → Level 16

Level Goal:

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.



```
bandit14@bandit:~$ ls
bandit14@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  .profile  .ssh
bandit14@bandit:~$ openssl s_client connect localhost:30001
s_client: Use -help for summary.
bandit14@bandit:~$ openssl s_client -connect localhost:30001
CONNECTED(00000003)
depth=0 CN = localhost
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = localhost
verify return:1
---
Certificate chain
 0 s:/CN=localhost
   i:/CN=localhost
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICBjCCAW+gAwIBAgIESUpi7DANBgkqhkiG9w0BAQUFADAUMRIwEAYDVQQDDAls
```



```
    TLS session ticket:
    0000 - 64 68 30 37 ad 56 84 7c-c1 99 6e d2 02 05 fa fe   dh07.V.|..n.....
    0010 - 51 56 56 d4 c8 cc 75 dc-b8 ad 5c c7 04 ac bc 80   QVV...u...\.....
    0020 - 51 96 06 4f cf ff 14 36-e5 14 9a bd ed 97 26 c6   Q..O...6......&.
    0030 - bd d7 80 5a 52 48 6a 41-ea b1 69 bd 62 81 68 b4   ...ZRHjA..i.b.h.
    0040 - 67 18 b1 f2 0a 1e ac 55-20 f8 ec ac 42 02 d2 b0   g......U ...B...
    0050 - 56 eb 83 06 0c 44 68 46-86 69 10 70 0b 23 94 5e   V....DhF.i.p.#.^
    0060 - 63 d6 77 1b 05 88 79 6d-e8 17 4c c9 e0 03 c5 69   c.w...ym..L....i
    0070 - a3 17 5f 96 3c 0a 2d fb-10 52 67 dd 9d cc a5 4c   .._.<.-..Rg....L
    0080 - bf 9c 0d 35 52 36 fa a8-a6 48 18 92 67 78 3e f2   ...5R6...H..gx>.
    0090 - 5d 35 0b 77 3e ac df c0-69 c7 2a fa d3 ff d4 92   ]5.w>...i.*.....

    Start Time: 1544615701
    Timeout   : 7200 (sec)
    Verify return code: 18 (self signed certificate)
    Extended master secret: yes
---
BfMYroe26WYalil77FoDi9qh59eK5xNr
Correct!
cluFn7wTiGryunymYOu4RcffSxQluehd

closed
bandit14@bandit:~$ 
```

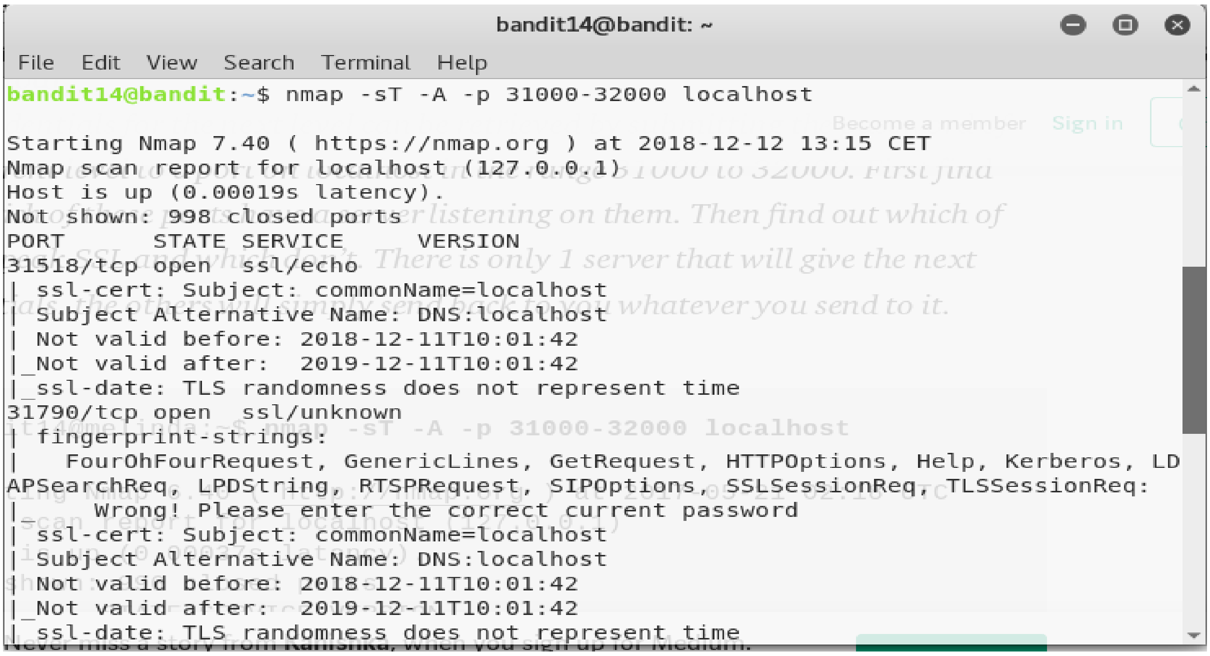
In this level, we can use SSL encryption, OpenSSL is a multi-platform, open source SSL/TLS toolkit, for c_client **"(-connect host:port)"**

Using s_connect, localhost as a username with the port number 30001, you can check the password. Password is:
cluFn7wTiGryunymYOu4RcffSxQluehd

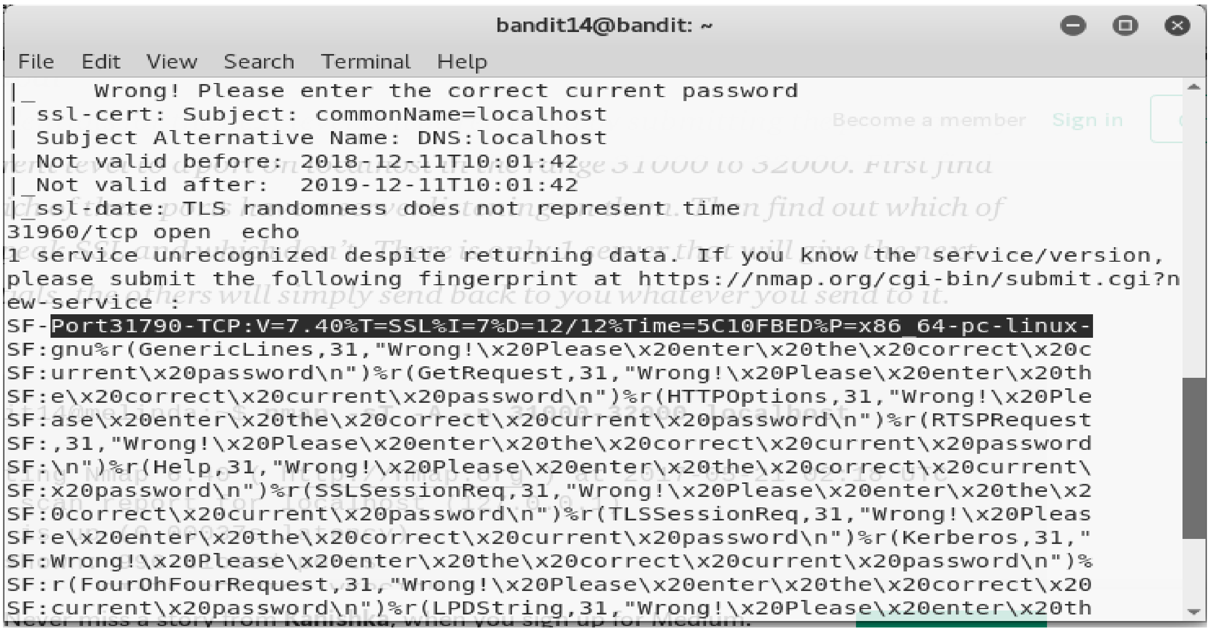
Bandit Level 16 → Level 17

Level Goal:

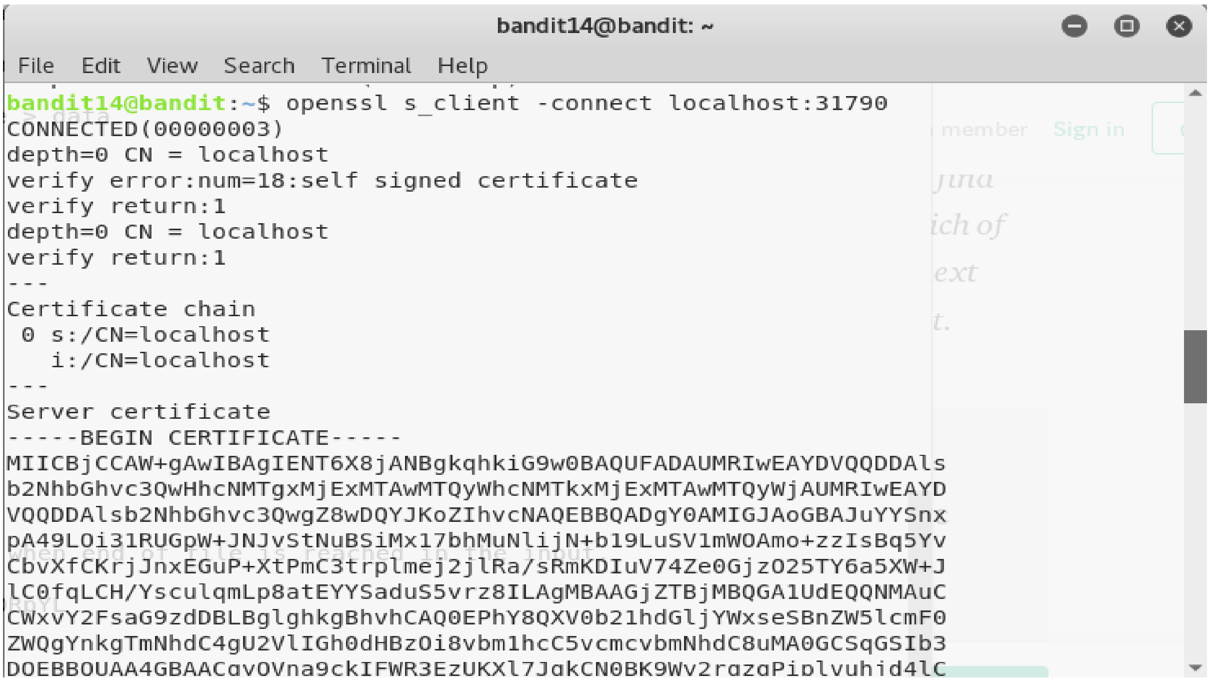
The credentials for the next level can be retrieved by submitting the password of the current level to **a port on localhost in the range 31000 to 32000**. First find out which of these ports have a server listening on them. Then find out which of those speak SSL and which don’t. There is only 1 server that will give the next credentials, the others will simply send back to you whatever you send to it.



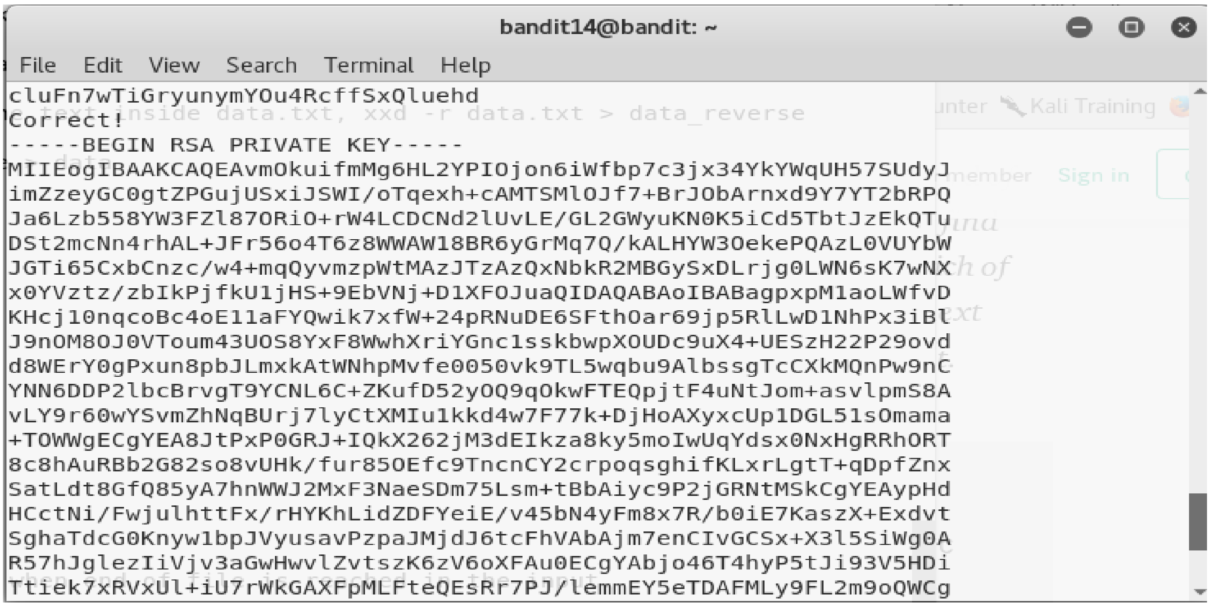
```
bandit14@bandit:~$ nmap -sT -A -p 31000-32000 localhost

Starting Nmap 7.40 ( https://nmap.org ) at 2018-12-12 13:15 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00019s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE     VERSION
31518/tcp open  ssl/echo
| ssl-cert: Subject: commonName=localhost
| Subject Alternative Name: DNS:localhost
| Not valid before: 2018-12-11T10:01:42
|_Not valid after:  2019-12-11T10:01:42
|_ssl-date: TLS randomness does not represent time
31790/tcp open  ssl/unknown
| fingerprint-strings:
|   FourOhFourRequest, GenericLines, GetRequest, HTTPOptions, Help, Kerberos, LD
APSearchReq, LPDString, RTSPRequest, SIPOptions, SSLSessionReq, TLSSessionReq:
|_    Wrong! Please enter the correct current password
| ssl-cert: Subject: commonName=localhost
| Subject Alternative Name: DNS:localhost
| Not valid before: 2018-12-11T10:01:42
|_Not valid after:  2019-12-11T10:01:42
|_ssl-date: TLS randomness does not represent time
```



```
|_    Wrong! Please enter the correct current password
| ssl-cert: Subject: commonName=localhost
| Subject Alternative Name: DNS:localhost
| Not valid before: 2018-12-11T10:01:42
|_Not valid after:  2019-12-11T10:01:42
|_ssl-date: TLS randomness does not represent time
31960/tcp open  echo
1 service unrecognized despite returning data. If you know the service/version,
please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?n
ew-service :
SF-Port31790-TCP:V=7.40%T=SSL%I=7%D=12/12%Time=5C10FBED%P=x86_64-pc-linux-
SF:gnu%r(GenericLines,31,"Wrong!\x20Please\x20enter\x20the\x20correct\x20c
SF:urrent\x20password\n")%r(GetRequest,31,"Wrong!\x20Please\x20enter\x20th
SF:e\x20correct\x20current\x20password\n")%r(HTTPOptions,31,"Wrong!\x20Ple
SF:ase\x20enter\x20the\x20correct\x20current\x20password\n")%r(RTSPRequest
SF:,31,"Wrong!\x20Please\x20enter\x20the\x20correct\x20current\x20password
SF:\n")%r(Help,31,"Wrong!\x20Please\x20enter\x20the\x20correct\x20current\
SF:x20password\n")%r(SSLSessionReq,31,"Wrong!\x20Please\x20enter\x20the\x2
SF:0correct\x20current\x20password\n")%r(TLSSessionReq,31,"Wrong!\x20Pleas
SF:e\x20enter\x20the\x20correct\x20current\x20password\n")%r(Kerberos,31,"
SF:Wrong!\x20Please\x20enter\x20the\x20correct\x20current\x20password\n")%
SF:r(FourOhFourRequest,31,"Wrong!\x20Please\x20enter\x20the\x20correct\x20
SF:current\x20password\n")%r(LPDString,31,"Wrong!\x20Please\x20enter\x20th
```



```
bandit14@bandit:~$ openssl s_client -connect localhost:31790
CONNECTED(00000003)
depth=0 CN = localhost
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = localhost
verify return:1
---
Certificate chain
 0 s:/CN=localhost
   i:/CN=localhost
---
Server certificate
-----BEGIN CERTIFICATE-----
MIICBjCCAW+gAwIBAgIENT6X8jANBgkqhkiG9w0BAQUFADAUMRIwEAYDVQQDDAls
b2NhbGhvc3QwHhcNMTgxMjExMTAwMTQyWhcNMTkxMjExMTAwMTQyWjAUMRIwEAYD
VQQDDAlsb2NhbGhvc3QwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAJuYYSnx
pA49LOi31RUGpW+JNJvStNuBSiMx17bhMuNlijN+b19LuSV1mWOAmo+zzIsBq5Yv
CbvXfCKrjJnxEGuP+XtPmC3trplmej2jlRa/sRmKDIuV74Ze0GjzO25TY6a5XW+J
lC0fqLCH/YsculqmLp8atEYYSaduS5vrz8ILAgMBAAGjZTBjMBQGA1UdEQQNMAuC
CWxvY2FsaG9zdDBLBglghkgBhvhCAQ0EPhY8QXV0b21hdGljYWxseSBnZW5lcmF0
ZWQgYnkgTmNhdC4gU2VlIGh0dHBzOi8vbm1hcC5vcmcvbmNhdC8uMA0GCSqGSIb3
DQEBBQUAA4GBAACqvOVna9ckIFWR3EzUKXl7JqkCN0BK9Wv2rqzqPiplvuhid4lC
```



```
cluFn7wTiGryunymYOu4RcffSxQluehd
Correct!
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAvmOkuifmMg6HL2YPIOjon6iWfbp7c3jx34YkYWqUH57SUdyJ
imZzeyGC0gtZPGujUSxiJSWI/oTqexh+cAMTSMlOJf7+BrJObArnxd9Y7YT2bRPQ
Ja6Lzb558YW3FZl87ORiO+rW4LCDCNd2lUvLE/GL2GWyuKN0K5iCd5TbtJzEkQTu
DSt2mcNn4rhAL+JFr56o4T6z8WWAW18BR6yGrMq7Q/kALHYW3OekePQAzL0VUYbW
JGTi65CxbCnzc/w4+mqQyvmzpWtMAzJTzAzQxNbkR2MBGySxDLrjg0LWN6sK7wNX
x0YVztz/zbIkPjfkU1jHS+9EbVNj+D1XFOJuaQIDAQABAoIBABagpxpM1aoLWfvD
KHcj10nqcoBc4oE11aFYQwik7xfW+24pRNuDE6SFthOar69jp5RlLwD1NhPx3iBl
J9nOM8OJ0VToum43UOS8YxF8WwhXriYGnc1sskbwpXOUDc9uX4+UESzH22P29ovd
d8WErY0gPxun8pbJLmxkAtWNhpMvfe0050vk9TL5wqbu9AlbssgTcCXkMQnPw9nC
YNN6DDP2lbcBrvgT9YCNL6C+ZKufD52yOQ9qOkwFTEQpjtF4uNtJom+asvlpmS8A
vLY9r60wYSvmZhNqBUrj7lyCtXMIu1kkd4w7F77k+DjHoAXyxcUp1DGL51sOmama
+TOWWgECgYEA8JtPxP0GRJ+IQkX262jM3dEIkza8ky5moIwUqYdsx0NxHgRRhORT
8c8hAuRBb2G82so8vUHk/fur85OEfc9TncnCY2crpoqsghifKLxrLgtT+qDpfZnx
SatLdt8GfQ85yA7hnWWJ2MxF3NaeSDm75Lsm+tBbAiyc9P2jGRNtMSkCgYEAypHd
HCctNi/FwjulhttFx/rHYKhLidZDFYeiE/v45bN4yFm8x7R/b0iE7KaszX+Exdvt
SghaTdcG0Knyw1bpJVyusavPzpaJMjdJ6tcFhVAbAjm7enCIvGCSx+X3l5SiWg0A
R57hJglezIiVjv3aGwHwvlZvtszK6zV6oXFAu0ECgYAbjo46T4hyP5tJi93V5HDi
Ttiek7xRVxUl+iU7rWkGAXFpMLFteQEsRr7PJ/lemmEY5eTDAFMLy9FL2m9oQWCg
```

For this level, you must learn about nmap : Network Mapper is open source and very versatile tool for Linux systems/network administrators. mainly used for network audit , performing security scans.

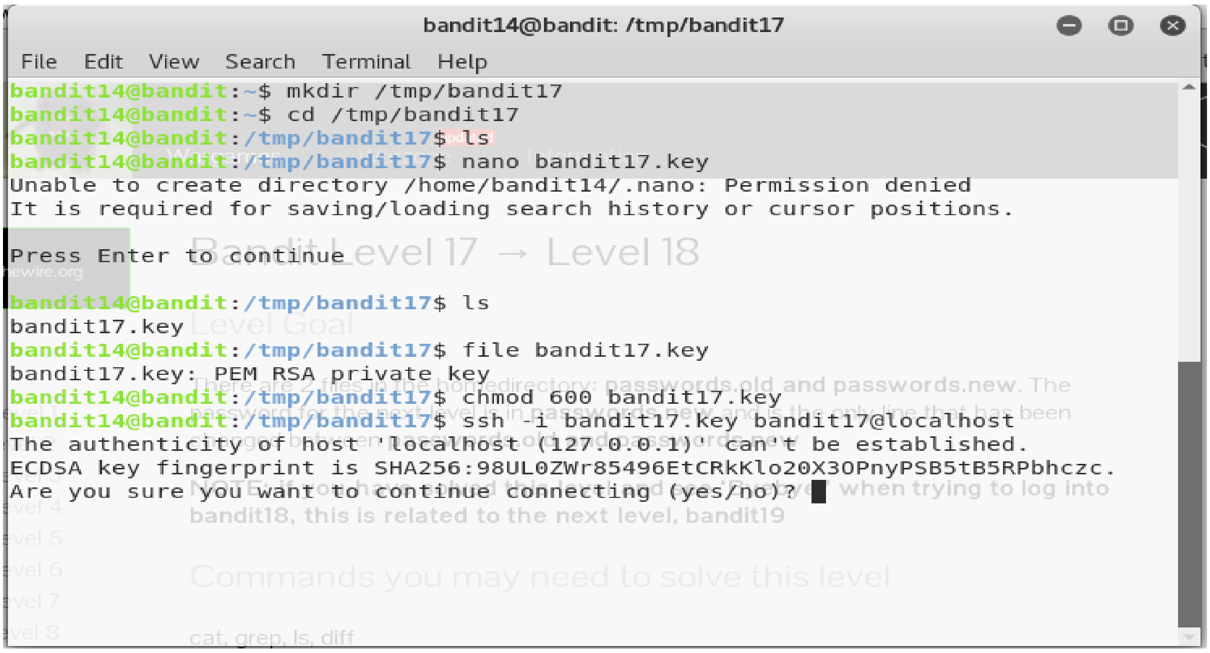
Between host number 31000-32000, only one port number give credentials and you can see the port number:31790 is having credentials,

You can connect with host number:31790, it will display RSA key certificate, you can use your password and it will display RSA Private Key. This key used for login **Level17.**

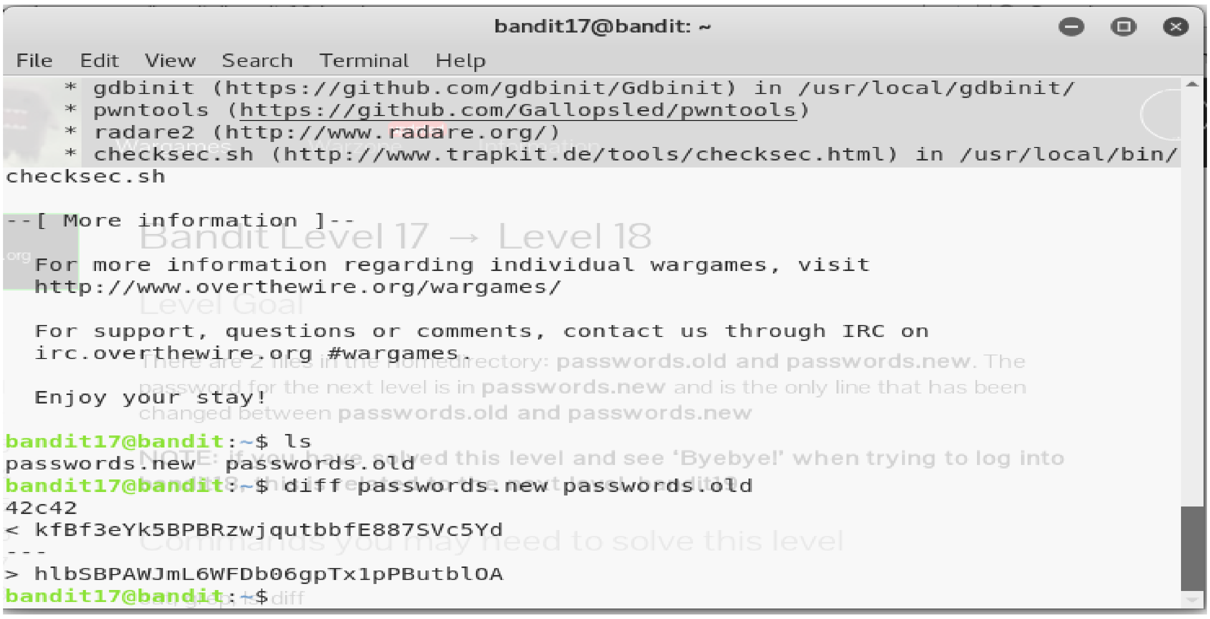
Bandit Level 17 → Level 18

Level Goal

There are 2 files in the home directory: **passwords.old and passwords.new**. The password for the next level is in **passwords.new** and is the only line that has been changed between **passwords.old and passwords.new**


bandit14@bandit: /tmp/bandit17
File Edit View Search Terminal Help

```
bandit14@bandit:~$ mkdir /tmp/bandit17
bandit14@bandit:~$ cd /tmp/bandit17
bandit14@bandit:/tmp/bandit17$ ls
bandit14@bandit:/tmp/bandit17$ nano bandit17.key
Unable to create directory /home/bandit14/.nano: Permission denied
It is required for saving/loading search history or cursor positions.

Press Enter to continue

bandit14@bandit:/tmp/bandit17$ ls
bandit17.key
bandit14@bandit:/tmp/bandit17$ file bandit17.key
bandit17.key: PEM RSA private key
bandit14@bandit:/tmp/bandit17$ chmod 600 bandit17.key
bandit14@bandit:/tmp/bandit17$ ssh -i bandit17.key bandit17@localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)?
```



```
    * gdbinit (https://github.com/gdbinit/Gdbinit) in /usr/local/gdbinit/
    * pwntools (https://github.com/Gallopsled/pwntools)
    * radare2 (http://www.radare.org/)
    * checksec.sh (http://www.trapkit.de/tools/checksec.html) in /usr/local/bin/
checksec.sh

--[ More information ]--

  For more information regarding individual wargames, visit
  http://www.overthewire.org/wargames/

  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit17@bandit:~$ ls
passwords.new  passwords.old
bandit17@bandit:~$ diff passwords.new passwords.old
42c42
< kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd
---
> hlbSBPAWJmL6WFDb06gpTx1pPButblOA
bandit17@bandit:~$
```

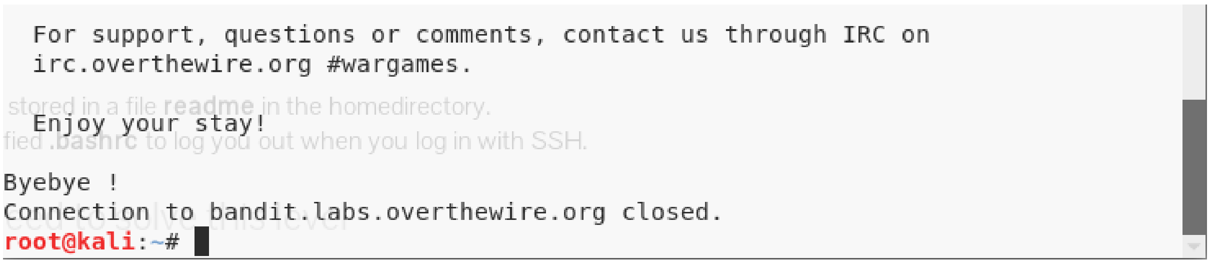
Here, we can see passwords and use both password for level 18 , it will work only one password, for this level, you must create a directory inside **/tmp/bandit17=> bandit17.key**

Next password is : **kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd**

Bandit Level 18 → Level 19

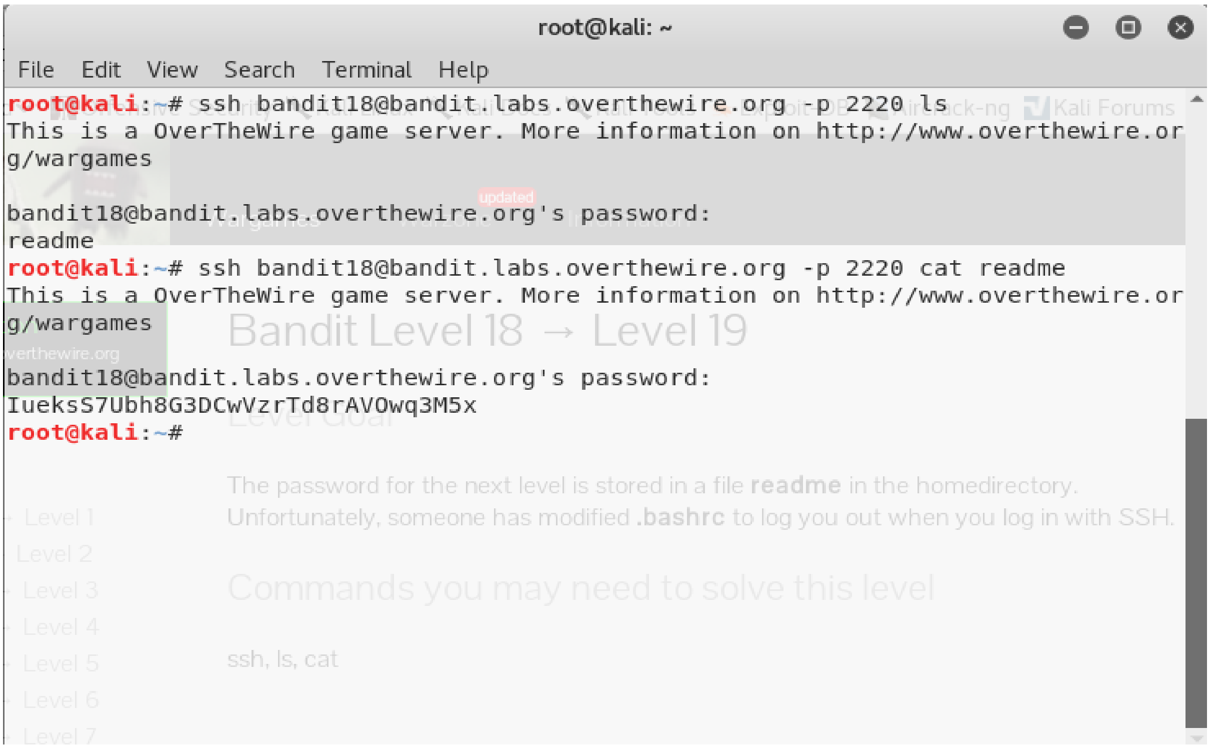
Level Goal

The password for the next level is stored in a file **readme** in the home directory. Unfortunately, someone has modified **.bashrc** to log you out when you log in with SSH.



```
  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

Byebye !
Connection to bandit.labs.overthewire.org closed.
root@kali:~# 
```

As I already shown in level 17-18 , that when I succeed level 17 and If I login for bandit18 using ssh command, you can see Byebye !.



```
root@kali:~# ssh bandit18@bandit.labs.overthewire.org -p 2220 ls
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

bandit18@bandit.labs.overthewire.org's password:
readme
root@kali:~# ssh bandit18@bandit.labs.overthewire.org -p 2220 cat readme
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

bandit18@bandit.labs.overthewire.org's password:
IueksS7Ubh8G3DCwVzrTd8rAVOwq3M5x
root@kali:~#
```

As already shown in this level that next level password is stored inside readme file, it is fine.

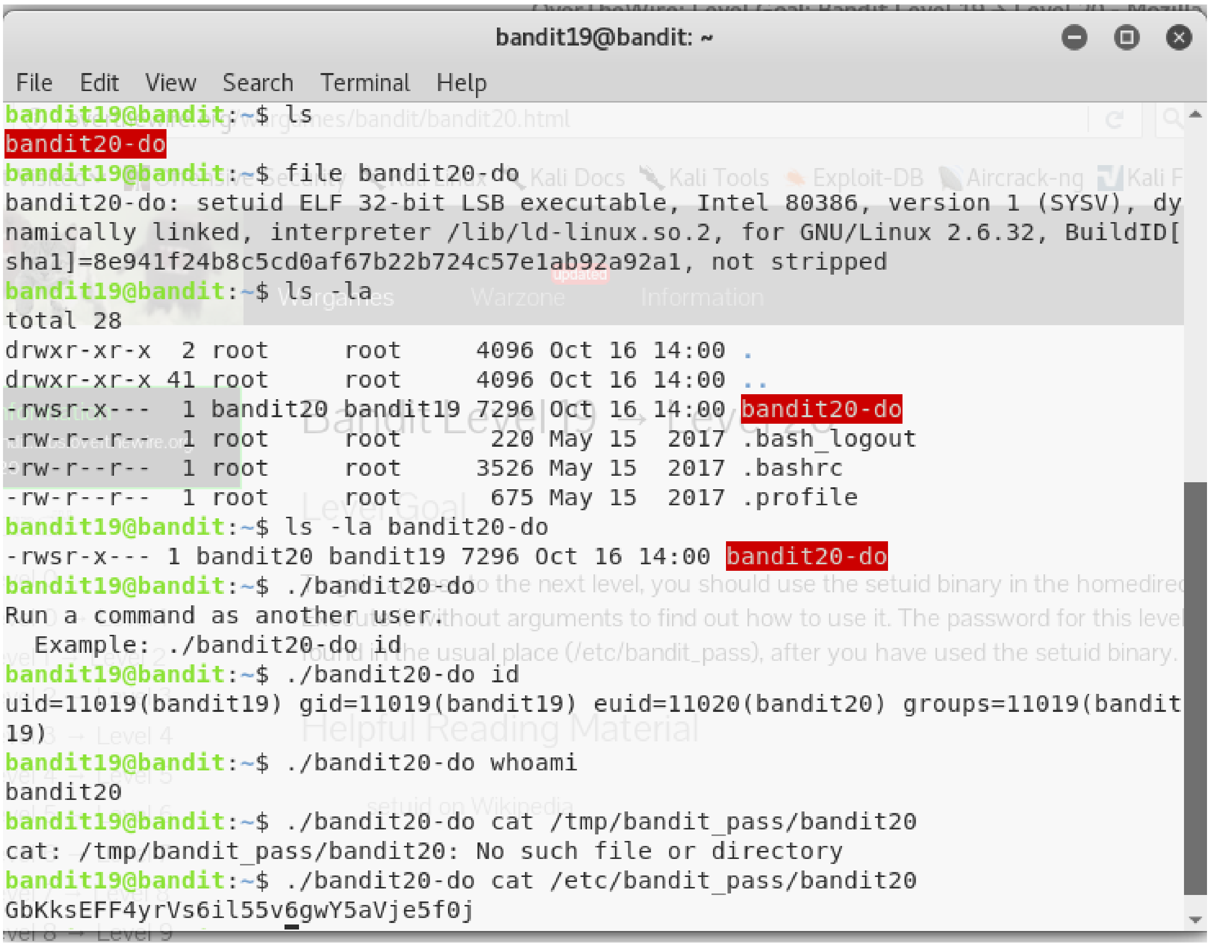
But unfortunately, someone has modified in **.bashrc**, when you ssh for connecting to next level, use **ls** command it will shows readme file and use **cat readme**, it will show Password for next level.

Password for the next level is: **IueksS7Ubh8G3DCwVzrTd8rAVOwq3M5x**

Bandit Level 19 → Level 20

Level Goal:

To gain access to the next level, you should use the setuid binary in the home directory. Execute it without arguments to find out how to use it. The password for this level can be found in the usual place (/etc/bandit_pass), after you have used the setuid binary.



```
bandit19@bandit:~$ ls
bandit20-do
bandit19@bandit:~$ file bandit20-do
bandit20-do: setuid ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dy
namically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.32, BuildID[
sha1]=8e941f24b8c5cd0af67b22b724c57e1ab92a92a1, not stripped
bandit19@bandit:~$ ls -la
total 28
drwxr-xr-x  2 root     root     4096 Oct 16 14:00 .
drwxr-xr-x 41 root     root     4096 Oct 16 14:00 ..
-rwsr-x---  1 bandit20 bandit19 7296 Oct 16 14:00 bandit20-do
-rw-r--r--  1 root     root      220 May 15  2017 .bash_logout
-rw-r--r--  1 root     root     3526 May 15  2017 .bashrc
-rw-r--r--  1 root     root      675 May 15  2017 .profile
bandit19@bandit:~$ ls -la bandit20-do
-rwsr-x--- 1 bandit20 bandit19 7296 Oct 16 14:00 bandit20-do
bandit19@bandit:~$ ./bandit20-do
Run a command as another user.
  Example: ./bandit20-do id
bandit19@bandit:~$ ./bandit20-do id
uid=11019(bandit19) gid=11019(bandit19) euid=11020(bandit20) groups=11019(bandit
19)
bandit19@bandit:~$ ./bandit20-do whoami
bandit20
bandit19@bandit:~$ ./bandit20-do cat /tmp/bandit_pass/bandit20
cat: /tmp/bandit_pass/bandit20: No such file or directory
bandit19@bandit:~$ ./bandit20-do cat /etc/bandit_pass/bandit20
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
```

Useful Command **setuid:** setuid and setgid are UNIX access right flags, these are allowed user to run an executable with the permission of the executable’s owner or group respectively and change behaviour in directories.

We can search file filename, we can check the file, and, in our level, file is setuid and you can check the long list and inside /etc/bandit_pass/bandit20, we can find next password

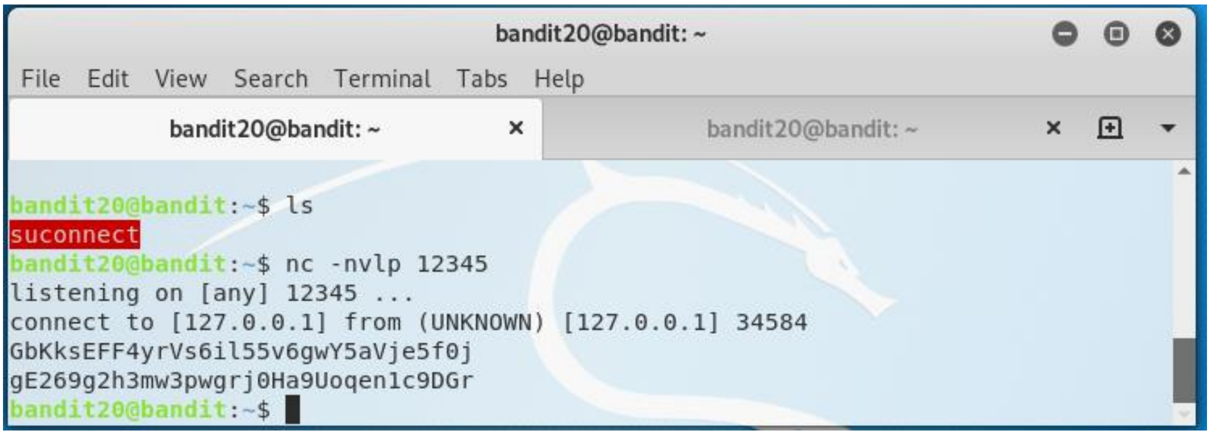
The password for the next Level is: **GbKksEFF4yrVs6il55v6gwY5aVje5f0j**

Bandit Level 20 → Level 21

Level Goal:

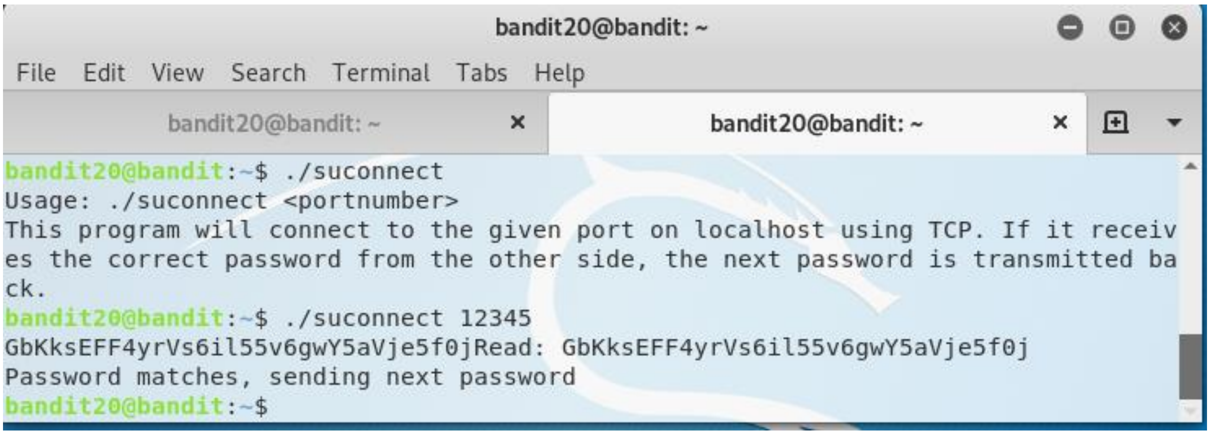
There is a setuid binary in the home directory that does the following: it makes a connection to localhost on the port you specify as a command line argument. It then reads a line of text from the connection and compares it to the password in the previous level (bandit20). If the password is correct, it will transmit the password for the next level.

Server Side:



```
bandit20@bandit:~$ ls
suconnect
bandit20@bandit:~$ nc -nvlp 12345
listening on [any] 12345 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 34584
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr
bandit20@bandit:~$
```

Client Side:



```
bandit20@bandit:~$ ./suconnect
Usage: ./suconnect <portnumber>
This program will connect to the given port on localhost using TCP. If it receiv
es the correct password from the other side, the next password is transmitted ba
ck.
bandit20@bandit:~$ ./suconnect 12345
GbKksEFF4yrVs6il55v6gwY5aVje5f0jRead: GbKksEFF4yrVs6il55v6gwY5aVje5f0j
Password matches, sending next password
bandit20@bandit:~$
```

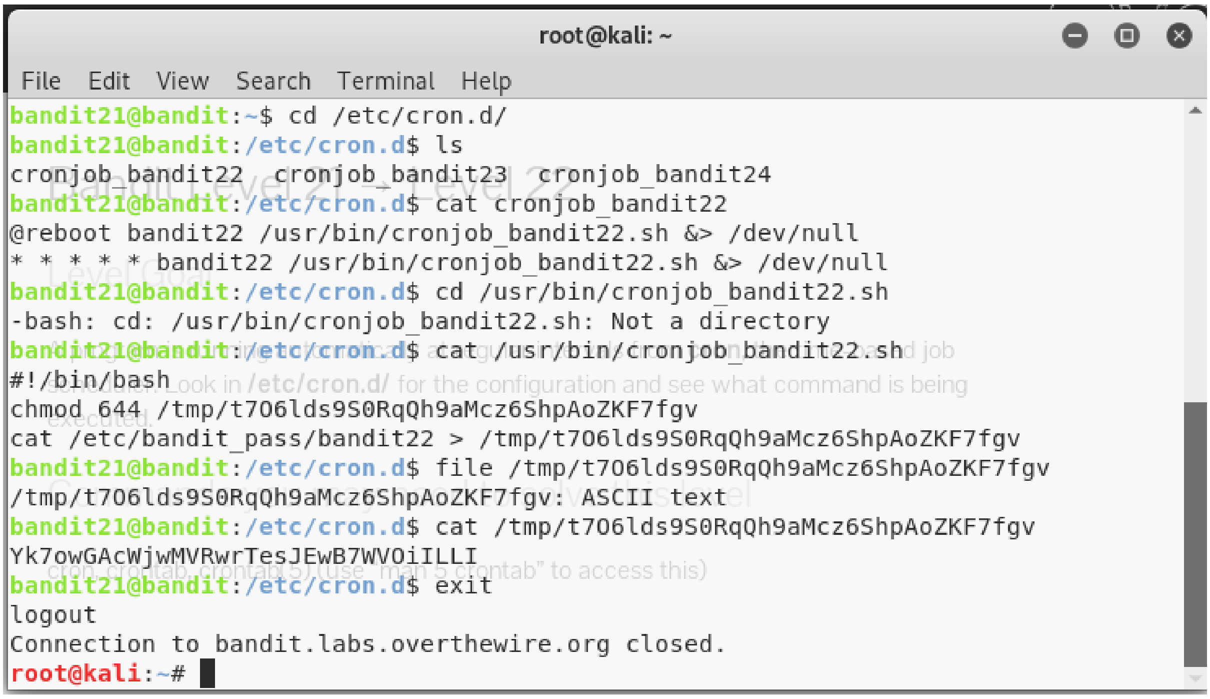
- On terminal 1 (host), start netcat as server: nc -nvlp 44444.
- On terminal 2 (client/setuid), run suconnect on same port: ./suconnect 44444.
- Back to terminal 1, send the password of current level.
- Suconnect will show you the password.

Password for the next Level is: **gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr**

Bandit Level 21 → Level 22

Level Goal

A program is running automatically at regular intervals from **cron**, the time-based job scheduler. Look in **/etc/cron.d/** for the configuration and see what command is being executed.



```
bandit21@bandit:~$ cd /etc/cron.d/
bandit21@bandit:/etc/cron.d$ ls
cronjob_bandit22  cronjob_bandit23  cronjob_bandit24
bandit21@bandit:/etc/cron.d$ cat cronjob_bandit22
@reboot bandit22 /usr/bin/cronjob_bandit22.sh &> /dev/null
* * * * * bandit22 /usr/bin/cronjob_bandit22.sh &> /dev/null
bandit21@bandit:/etc/cron.d$ cd /usr/bin/cronjob_bandit22.sh
-bash: cd: /usr/bin/cronjob_bandit22.sh: Not a directory
bandit21@bandit:/etc/cron.d$ cat /usr/bin/cronjob_bandit22.sh
#!/bin/bash
chmod 644 /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
cat /etc/bandit_pass/bandit22 > /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
bandit21@bandit:/etc/cron.d$ file /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
/tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv: ASCII text
bandit21@bandit:/etc/cron.d$ cat /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI
bandit21@bandit:/etc/cron.d$ exit
logout
Connection to bandit.labs.overthewire.org closed.
root@kali:~#
```

For this level, you can check the hint he gave that configuration directory /etc/cron.d,

It contains some files and inside crinjpb_bandit22 seem to be might interested and it shows a location if cronjob_bandit22.sh script.

Someone is dumping the password of bandit22 into a tmp file. We once again cat the tmp file and find the next password.

Password for the next level is:

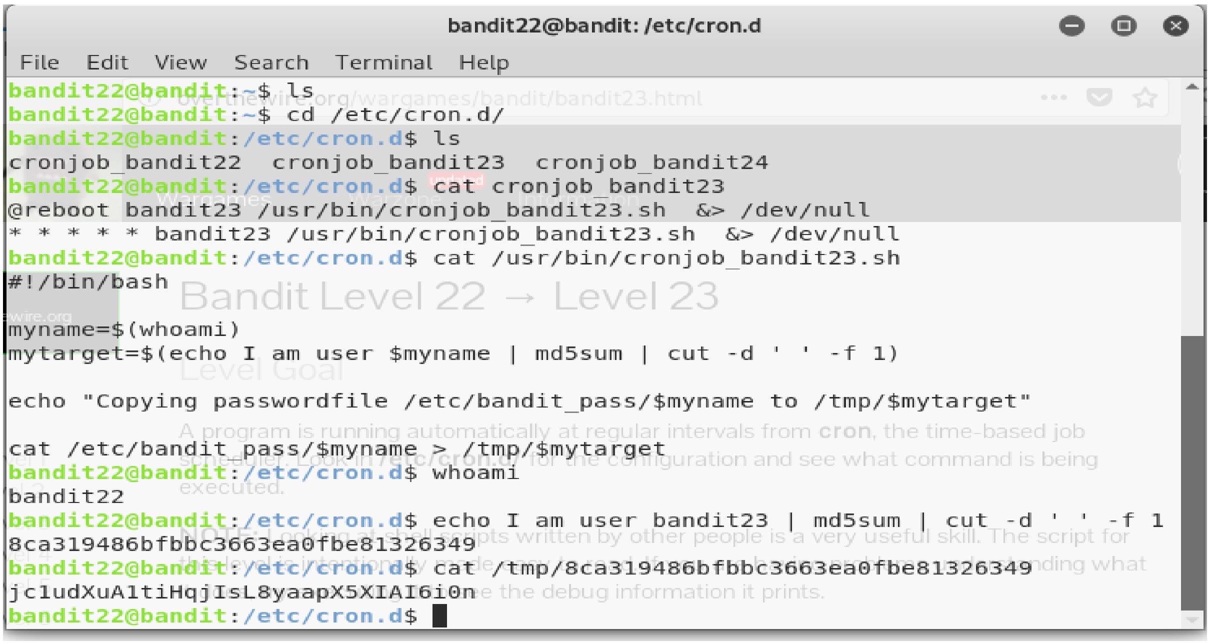
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI

Bandit Level 22 → Level 23

Level Goal:

A program is running automatically at regular intervals from **cron**, the time-based job scheduler. Look in **/etc/cron.d/** for the configuration and see what command is being executed.

NOTE: Looking at shell scripts written by other people is a very useful skill. The script for this level is intentionally made easy to read. If you are having problems understanding what it does, try executing it to see the debug information it prints.


bandit22@bandit: /etc/cron.d
File Edit View Search Terminal Help

```
bandit22@bandit:~$ ls
bandit22@bandit:~$ cd /etc/cron.d/
bandit22@bandit:/etc/cron.d$ ls
cronjob_bandit22  cronjob_bandit23  cronjob_bandit24
bandit22@bandit:/etc/cron.d$ cat cronjob_bandit23
@reboot bandit23 /usr/bin/cronjob_bandit23.sh  &> /dev/null
* * * * * bandit23 /usr/bin/cronjob_bandit23.sh  &> /dev/null
bandit22@bandit:/etc/cron.d$ cat /usr/bin/cronjob_bandit23.sh
#!/bin/bash

myname=$(whoami)
mytarget=$(echo I am user $myname | md5sum | cut -d ' ' -f 1)

echo "Copying passwordfile /etc/bandit_pass/$myname to /tmp/$mytarget"

cat /etc/bandit_pass/$myname > /tmp/$mytarget
bandit22@bandit:/etc/cron.d$ whoami
bandit22
bandit22@bandit:/etc/cron.d$ echo I am user bandit23 | md5sum | cut -d ' ' -f 1
8ca319486bfbbc3663ea0fbe81326349
bandit22@bandit:/etc/cron.d$ cat /tmp/8ca319486bfbbc3663ea0fbe81326349
jc1udXuA1tiHqjIsL8yaapX5XIAI6i0n
bandit22@bandit:/etc/cron.d$ 
```

We can open first /etc/cron.d it contains some files and in that cronjob_bandit23 seems to different because we already checked the bandit22 file,

It is showing one more path and open it using cat command , you can see the first bash script.

We got long string and looking at the content of this file in tmp folder gives us the next password.

The password for the next password is : **jc1udXuA1tiHqjIsL8yaapX5XIAI6i0n**

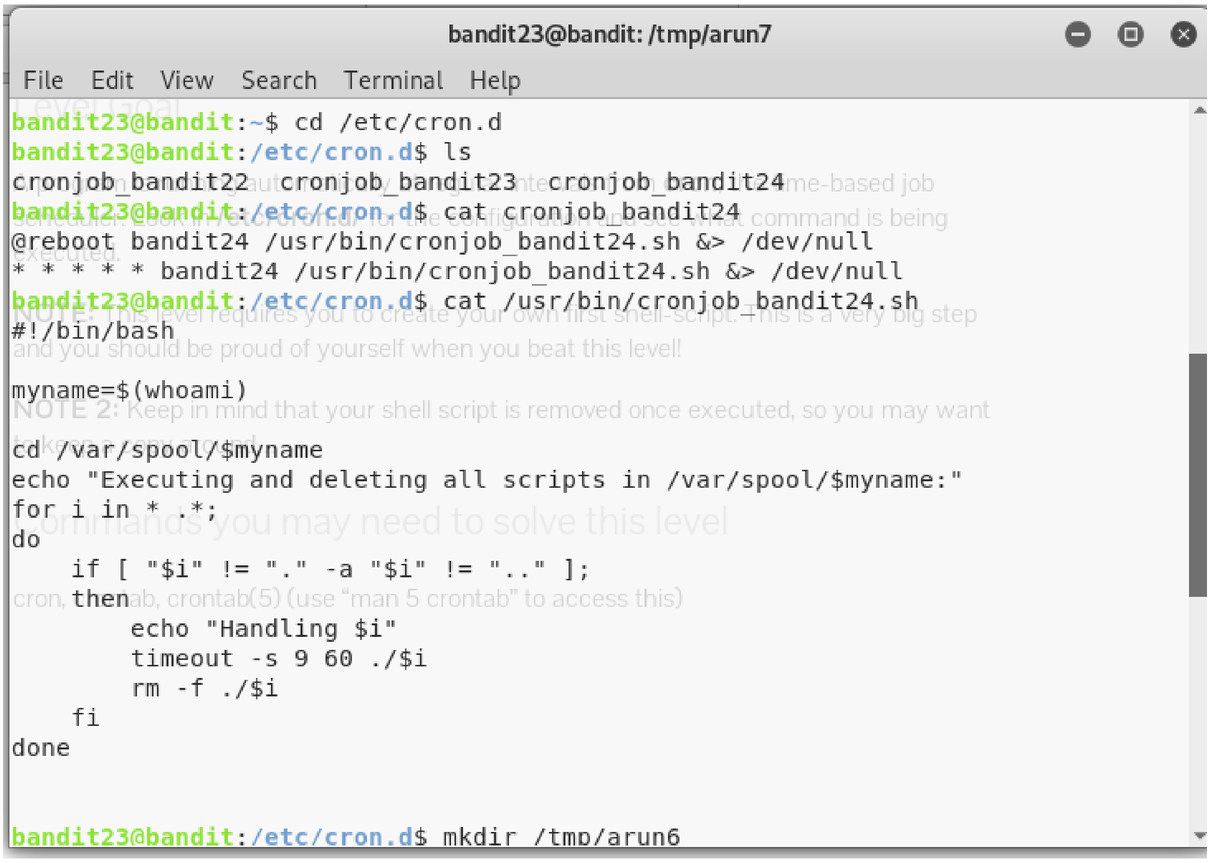
Bandit Level 23 → Level 24

Level Goal

A program is running automatically at regular intervals from **cron**, the time-based job scheduler. Look in **/etc/cron.d/** for the configuration and see what command is being executed.

NOTE: This level requires you to create your own first shell-script. This is a very big step and you should be proud of yourself when you beat this level!

NOTE 2: Keep in mind that your shell script is removed once executed, so you may want to keep a copy around…


bandit23@bandit: /tmp/arun7
File Edit View Search Terminal Help

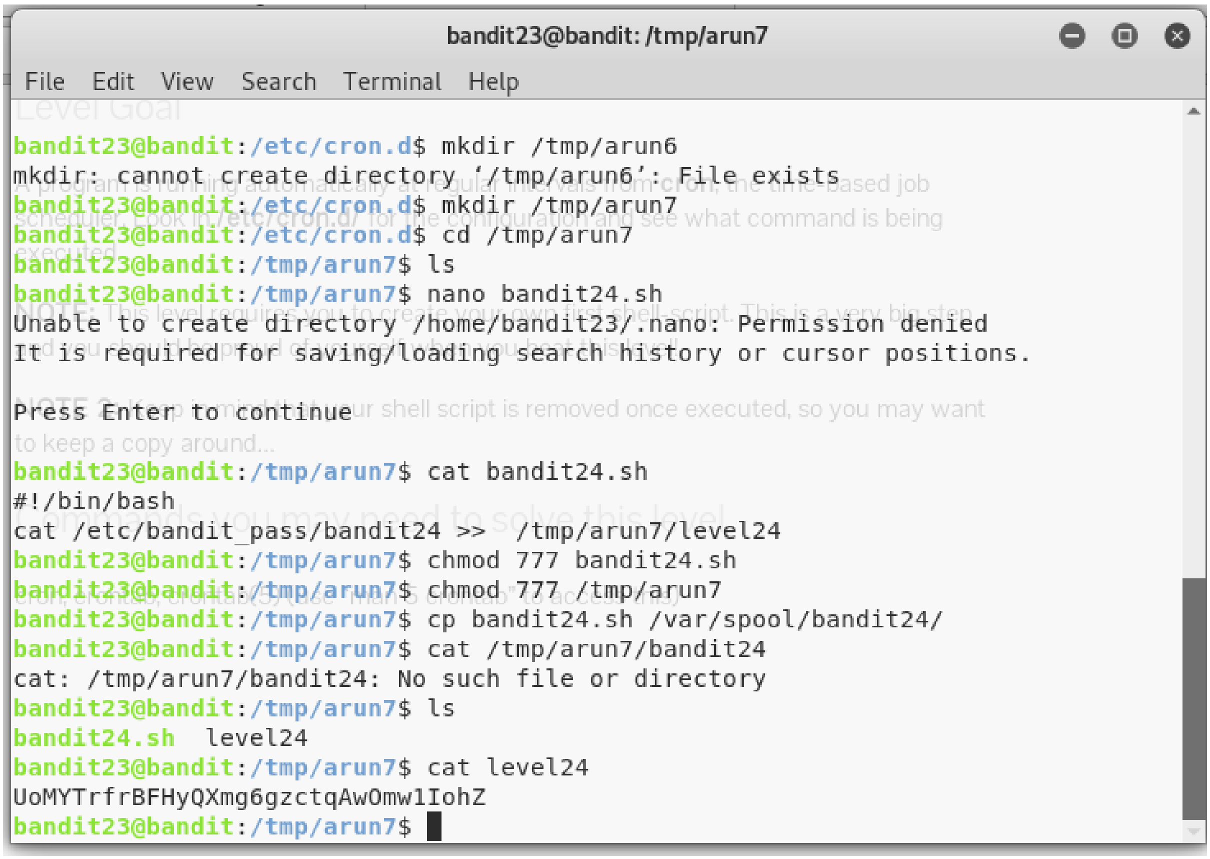
```
bandit23@bandit:~$ cd /etc/cron.d
bandit23@bandit:/etc/cron.d$ ls
cronjob_bandit22  cronjob_bandit23  cronjob_bandit24
bandit23@bandit:/etc/cron.d$ cat cronjob_bandit24
@reboot bandit24 /usr/bin/cronjob_bandit24.sh &> /dev/null
* * * * * bandit24 /usr/bin/cronjob_bandit24.sh &> /dev/null
bandit23@bandit:/etc/cron.d$ cat /usr/bin/cronjob_bandit24.sh
#!/bin/bash

myname=$(whoami)

cd /var/spool/$myname
echo "Executing and deleting all scripts in /var/spool/$myname:"
for i in * .*;
do
    if [ "$i" != "." -a "$i" != ".." ];
    then
        echo "Handling $i"
        timeout -s 9 60 ./$i
        rm -f ./$i
    fi
done


bandit23@bandit:/etc/cron.d$ mkdir /tmp/arun6
```

We can open the /etc/cron.d path, we found some files like last 2 levels and in that cronjob_bandit24 is interesting and you will get a path. Using cat command, you can enter cronjob_bandit24.sh, we can see the first bash script.

From the description of the script, it will execute all the script inside the $myname folder. We found that there is a bandit24 folder in /var/spool/.

Therefore, let's get a simple script of copying the password to a tmp folder (like two levels before)

At this point, I can copy the file, to /var/spool/bandit24/ but I remember the permission for execute must be set.


bandit23@bandit: /tmp/arun7
File Edit View Search Terminal Help

```
bandit23@bandit:/etc/cron.d$ mkdir /tmp/arun6
mkdir: cannot create directory '/tmp/arun6': File exists
bandit23@bandit:/etc/cron.d$ mkdir /tmp/arun7
bandit23@bandit:/etc/cron.d$ cd /tmp/arun7
bandit23@bandit:/tmp/arun7$ ls
bandit23@bandit:/tmp/arun7$ nano bandit24.sh
Unable to create directory /home/bandit23/.nano: Permission denied
It is required for saving/loading search history or cursor positions.

Press Enter to continue
bandit23@bandit:/tmp/arun7$ cat bandit24.sh
#!/bin/bash
cat /etc/bandit_pass/bandit24 >> /tmp/arun7/level24
bandit23@bandit:/tmp/arun7$ chmod 777 bandit24.sh
bandit23@bandit:/tmp/arun7$ chmod 777 /tmp/arun7
bandit23@bandit:/tmp/arun7$ cp bandit24.sh /var/spool/bandit24/
bandit23@bandit:/tmp/arun7$ cat /tmp/arun7/bandit24
cat: /tmp/arun7/bandit24: No such file or directory
bandit23@bandit:/tmp/arun7$ ls
bandit24.sh  level24
bandit23@bandit:/tmp/arun7$ cat level24
UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ
bandit23@bandit:/tmp/arun7$
```

We must give permission to your bandit24.sh file using chmod command.

After that copy the bash file inside the /var/spool/bandit24

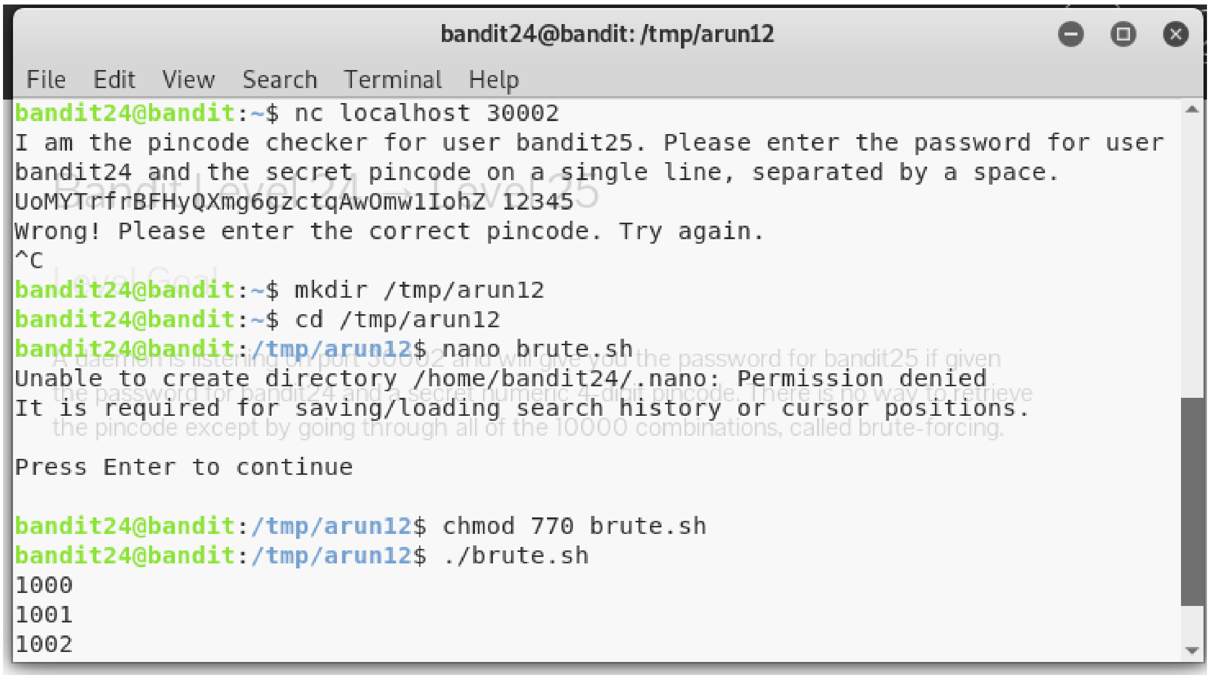
Now you can see inside the /tmp/arun7 , you can see the 2 files, using cat command open the level24 to see the password

Password for the next level is : **UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ**

Bandit Level 24 → Level 25

Level Goal

A daemon is listening on port 30002 and will give you the password for bandit25 if given the password for bandit24 and a secret numeric 4-digit pincode. There is no way to retrieve the pincode except by going through all of the 10000 combinations, called brute-forcing.


bandit24@bandit: /tmp/arun12
File Edit View Search Terminal Help

```
bandit24@bandit:~$ nc localhost 30002
I am the pincode checker for user bandit25. Please enter the password for user
bandit24 and the secret pincode on a single line, separated by a space.
UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ 12345
Wrong! Please enter the correct pincode. Try again.
^C
bandit24@bandit:~$ mkdir /tmp/arun12
bandit24@bandit:~$ cd /tmp/arun12
bandit24@bandit:/tmp/arun12$ nano brute.sh
Unable to create directory /home/bandit24/.nano: Permission denied
It is required for saving/loading search history or cursor positions.

Press Enter to continue

bandit24@bandit:/tmp/arun12$ chmod 770 brute.sh
bandit24@bandit:/tmp/arun12$ ./brute.sh
1000
1001
1002
```

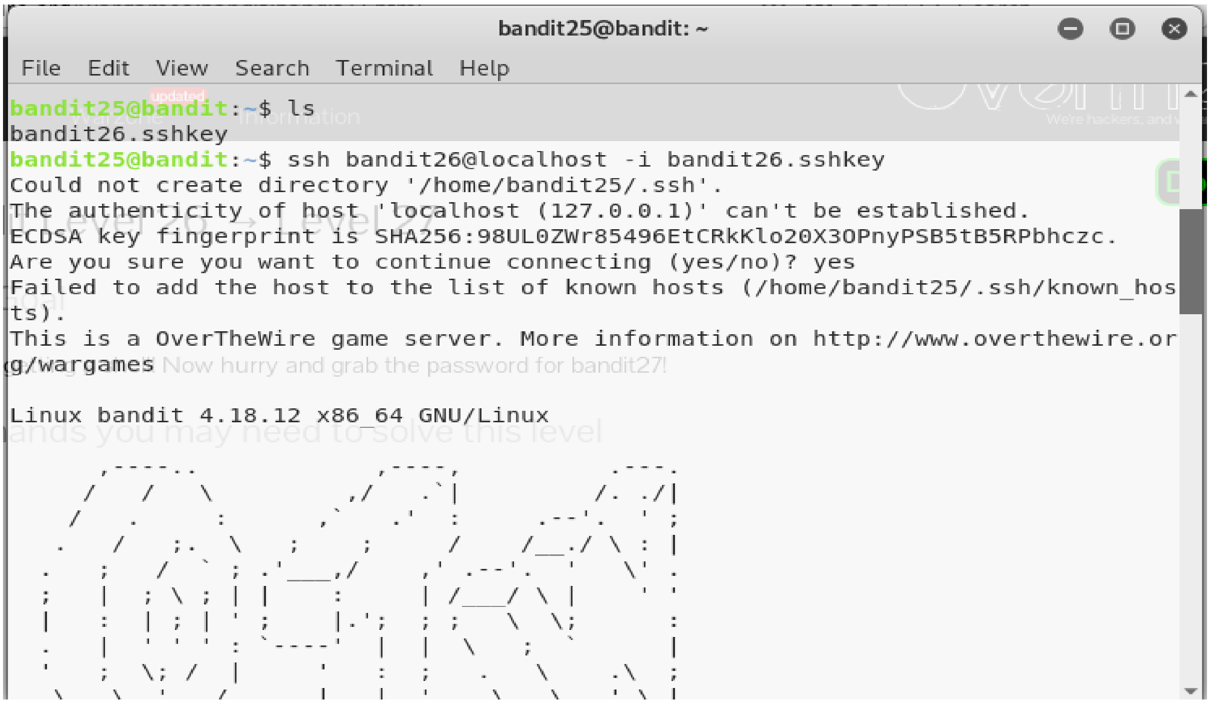
This level need lot of time to get a password, because we are using brute-force algorithm from 0000 to 10000,

For, this step we need a lot of patience and

Bandit Level 25 → Level 26

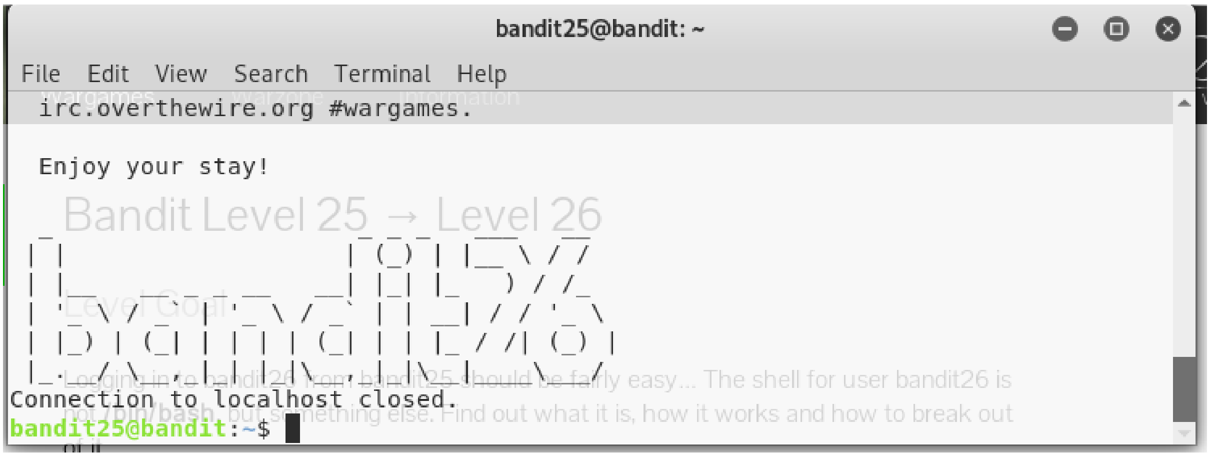
Level Goal

Logging in to bandit26 from bandit25 should be fairly easy… The shell for user bandit26 is not **/bin/bash**, but something else. Find out what it is, how it works and how to break out of it.



```
bandit25@bandit:~$ ls
bandit26.sshkey
bandit25@bandit:~$ ssh bandit26@localhost -i bandit26.sshkey
Could not create directory '/home/bandit25/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit25/.ssh/known_hos
ts).
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

Linux bandit 4.18.12 x86_64 GNU/Linux

        ,----..            ,----,          .---.
       /   /   \         ,/   .`|         /. ./|
      /   .     :      ,`   .'  :     .--'.  ' ;
     .   /   ;.  \   ;    ;     /    /__./ \ : |
    .   ;   /  ` ; .'___,/    ,' .--'.  '   \' .
    ;   |  ; \ ; | |    :     |  /__./ \ |    ' '
    |   :  | ; | ' ;    |.';  ;  ;    \  \;      :
    .   |  ' ' ' : `----'  |  |  \    ;  `      |
    '   ;  \; /  |         '   :  ;   .   \    .\  ;
     \   \  ',  /          |   |  '    \   \   ' \ |
```

For this role we have to login two times,



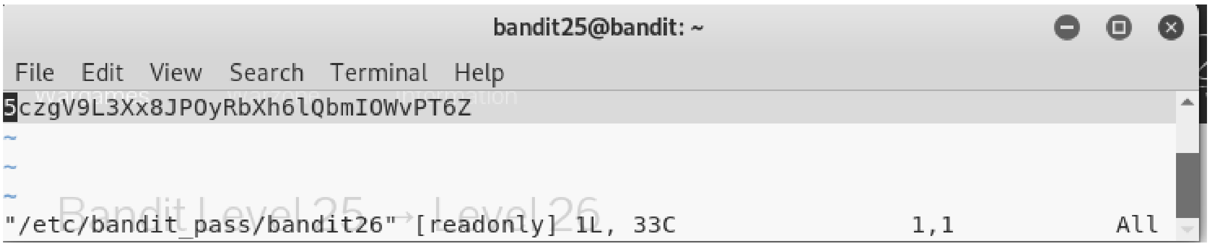
```
  irc.overthewire.org #wargames.

  Enjoy your stay!

  _                     _ _ _   ___   __
 | |                   | (_) | |__ \ / /
 | |__   __ _ _ __   __| |_| |_   ) / /_
 | '_ \ / _` | '_ \ / _` | | __| / / '_ \
 | |_) | (_| | | | | (_| | | |_ / /| (_) |
 |_.__/ \__,_|_| |_|\__,_|_|\__|____\___/
Connection to localhost closed.
bandit25@bandit:~$
```

However, after you logged into bandit26, you will be logged out immediately, “Connection to localhost closed.”

First, minimize your terminal so that when you are logged into bandit26 via ssh command, the large **“bandit26”** ASCII art banner will force a “more” message to prompt you to continue the output.

Now that you have forces the terminal to prompt you to continue the display via “more” or “–More–(50%)” in this case, press “v” to enter “vim”, a built-in text editor on Unix machines.

Use command **:e /etc/bandit_pass/bandit25** and press ENTER,



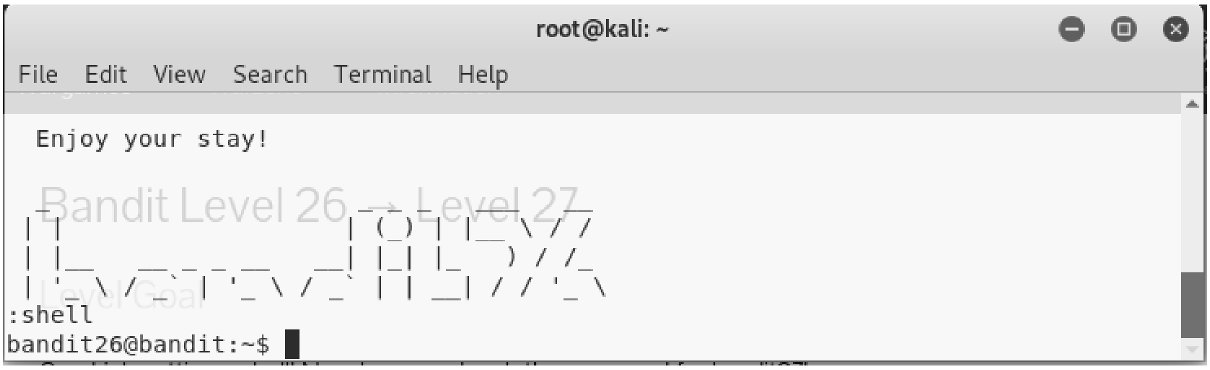
```
5czgV9L3Xx8JPOyRbXh6lQbmIOWvPT6Z
~
~
~
"/etc/bandit_pass/bandit26" [readonly] 1L, 33C                1,1           All
```

The password for next level is : **5czgV9L3Xx8JPOyRbXh6lQbmIOWvPT6Z**

Bandit Level 26 → Level 27

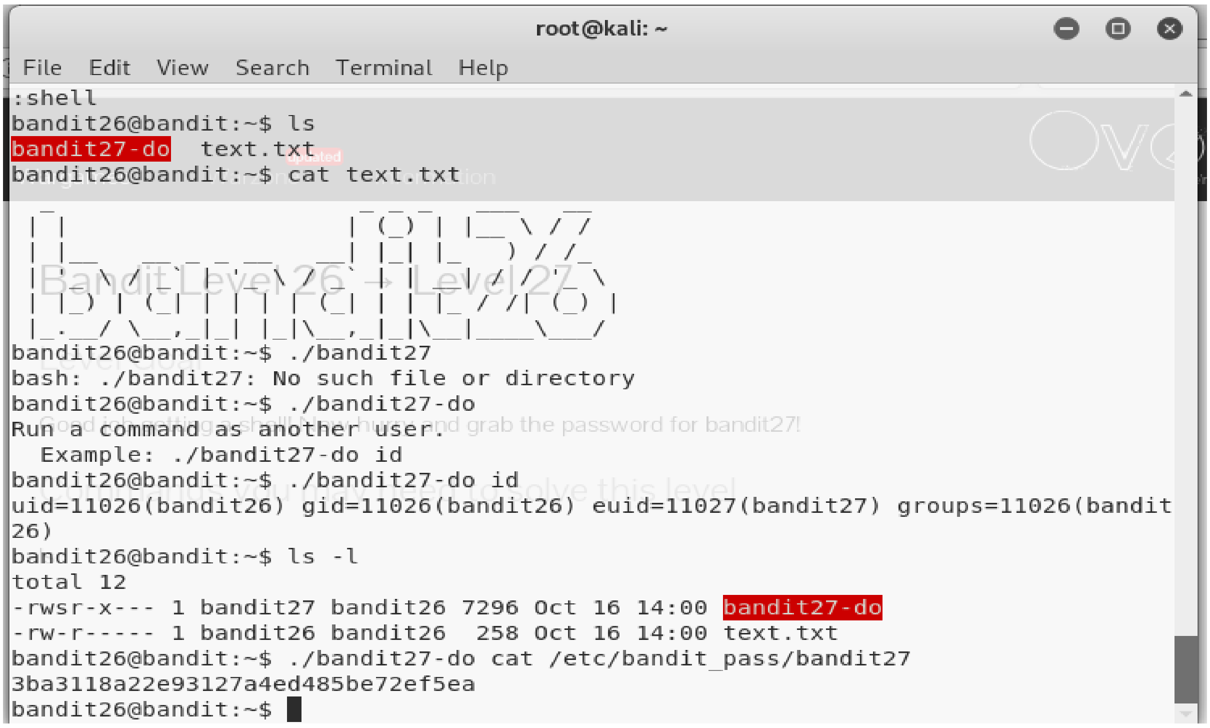
Level Goal

Good job getting a shell! Now hurry and grab the password for bandit27!



```
  Enjoy your stay!

  _                     _ _ _   ___   __
 | |                   | (_) | |__ \ / /
 | |__   __ _ _ __   __| |_| |_   ) / /_
 | '_ \ / _` | '_ \ / _` | | __| / / '_ \
:shell
bandit26@bandit:~$ 
```

Same as Level 26, login the level into 2 times. But here you can use first "V" for vim editor and use command :set shell /etc/bandit_pass/bandit24 press enter, again enter **:shell command** will give to allow to enter next level 26,



```
:shell
bandit26@bandit:~$ ls
bandit27-do  text.txt
bandit26@bandit:~$ cat text.txt
  _                     _ _ _   ___   __
 | |                   | (_) | |__ \ / /
 | |__   __ _ _ __   __| |_| |_   ) / /_
 | '_ \ / _` | '_ \ / _` | | __| / / '_ \
 | |_) | (_| | | | | (_| | | |_ / /| (_) |
 |_.__/ \__,_|_| |_|\__,_|_|\__|____\___/
bandit26@bandit:~$ ./bandit27
bash: ./bandit27: No such file or directory
bandit26@bandit:~$ ./bandit27-do
Run a command as another user.
  Example: ./bandit27-do id
bandit26@bandit:~$ ./bandit27-do id
uid=11026(bandit26) gid=11026(bandit26) euid=11027(bandit27) groups=11026(bandit
26)
bandit26@bandit:~$ ls -l
total 12
-rwsr-x--- 1 bandit27 bandit26 7296 Oct 16 14:00 bandit27-do
-rw-r----- 1 bandit26 bandit26  258 Oct 16 14:00 text.txt
bandit26@bandit:~$ ./bandit27-do cat /etc/bandit_pass/bandit27
3ba3118a22e93127a4ed485be72ef5ea
bandit26@bandit:~$ 
```

Now, use ls to check inside the list and you can find to files text.txt and bandit27-do.

Please follow above commands and you will get a next level password inside the bandit27-do file

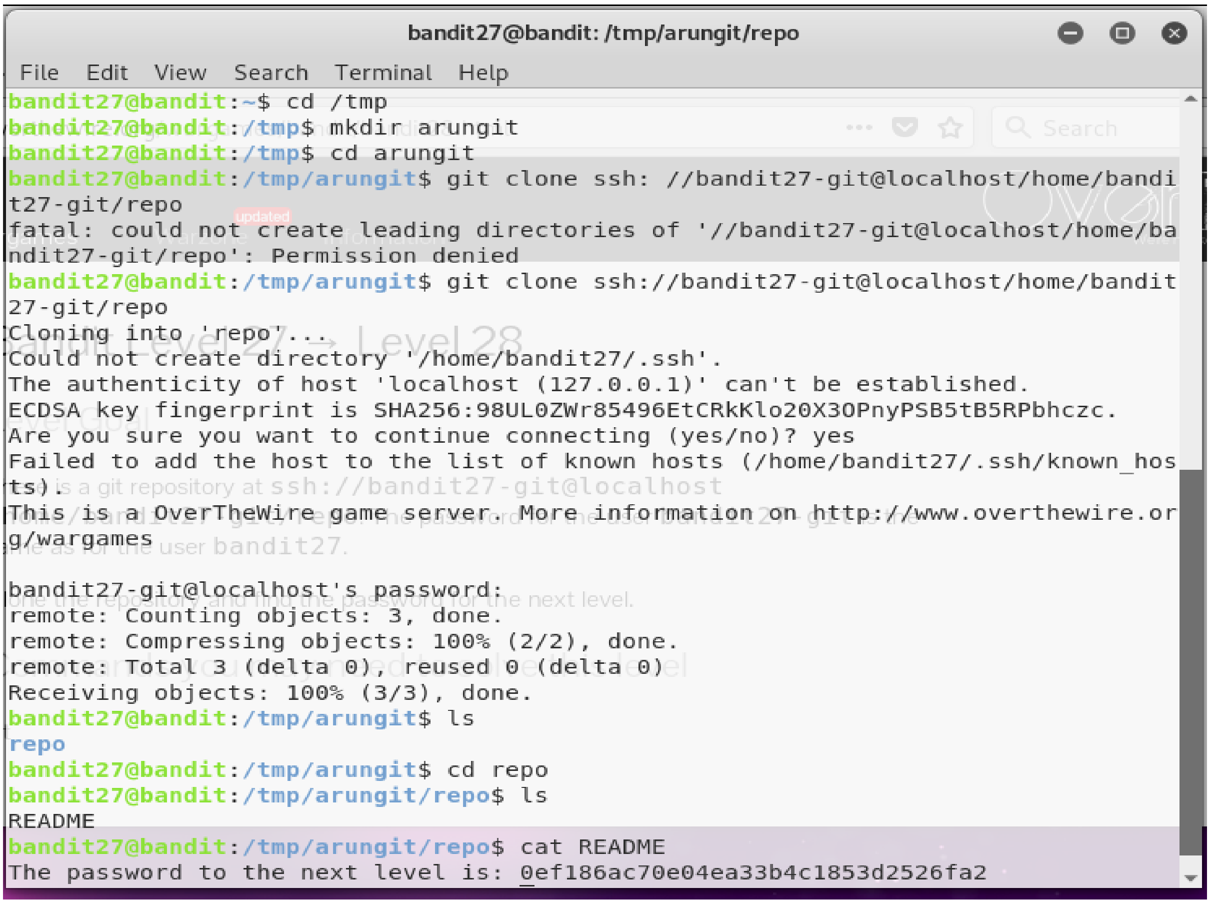
Password for the next level is: **3ba3118a22e93127a4ed485be72ef5ea**

Bandit Level 27 → Level 28

Level Goal

There is a git repository at ssh://bandit27-git@localhost/home/bandit27-git/repo. The password for the user bandit27-git is the same as for the user bandit27.

Clone the repository and find the password for the next level.


bandit27@bandit: /tmp/arungit/repo
File Edit View Search Terminal Help

```
bandit27@bandit:~$ cd /tmp
bandit27@bandit:/tmp$ mkdir arungit
bandit27@bandit:/tmp$ cd arungit
bandit27@bandit:/tmp/arungit$ git clone ssh: //bandit27-git@localhost/home/bandi
t27-git/repo
fatal: could not create leading directories of '//bandit27-git@localhost/home/ba
ndit27-git/repo': Permission denied
bandit27@bandit:/tmp/arungit$ git clone ssh://bandit27-git@localhost/home/bandit
27-git/repo
Cloning into 'repo'...
Could not create directory '/home/bandit27/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit27/.ssh/known_hos
ts).
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

bandit27-git@localhost's password:
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (3/3), done.
bandit27@bandit:/tmp/arungit$ ls
repo
bandit27@bandit:/tmp/arungit$ cd repo
bandit27@bandit:/tmp/arungit/repo$ ls
README
bandit27@bandit:/tmp/arungit/repo$ cat README
The password to the next level is: 0ef186ac70e04ea33b4c1853d2526fa2
```

Now we need to work with git.

Using **git clone** command, we receive an address to clone the repository through ssh.

Inside repo directory we can find one file is called README.

Use cat command to display the password.

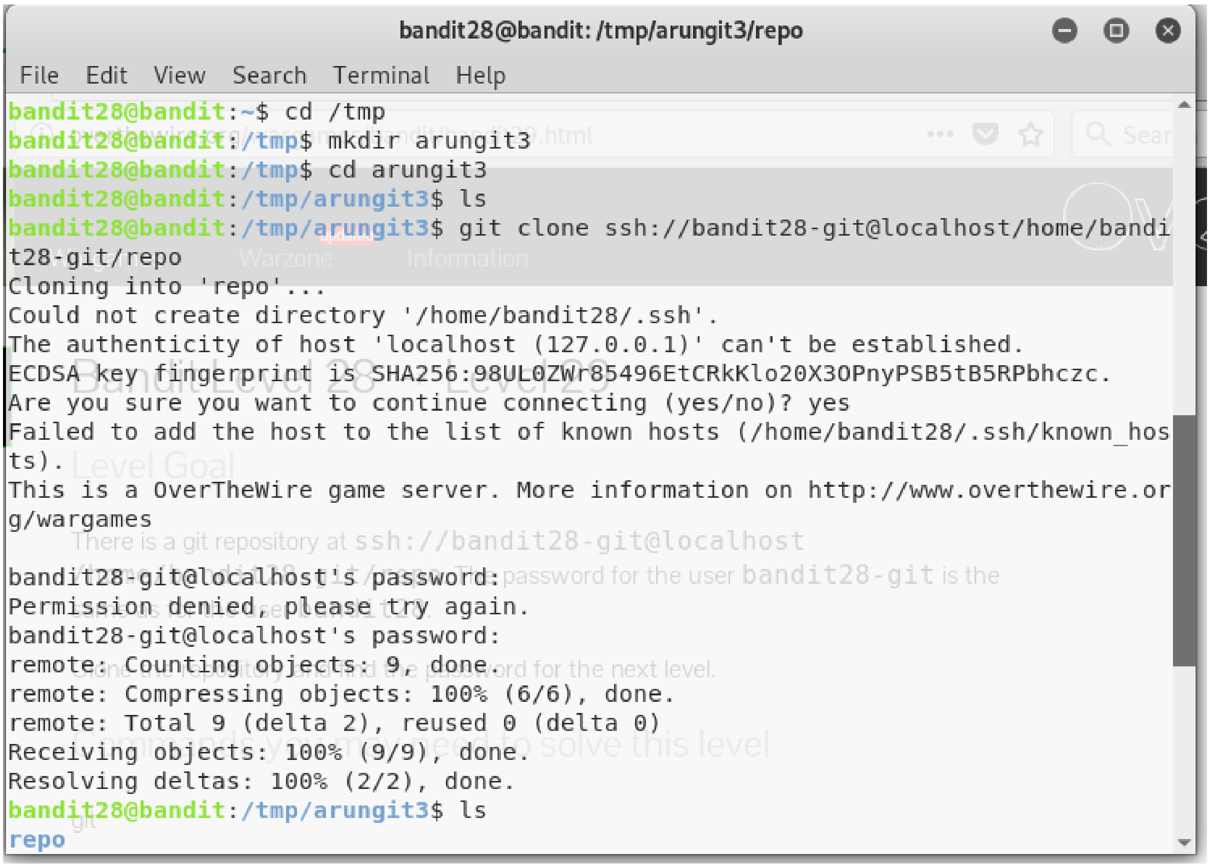
Password for the next level is : **0ef186ac70e04ea33b4c1853d2526fa2**

Bandit Level 28 → Level 29

Level Goal

There is a git repository at ssh://bandit28-git@localhost/home/bandit28-git/repo. The password for the user bandit28-git is the same as for the user bandit28.

Clone the repository and find the password for the next level.



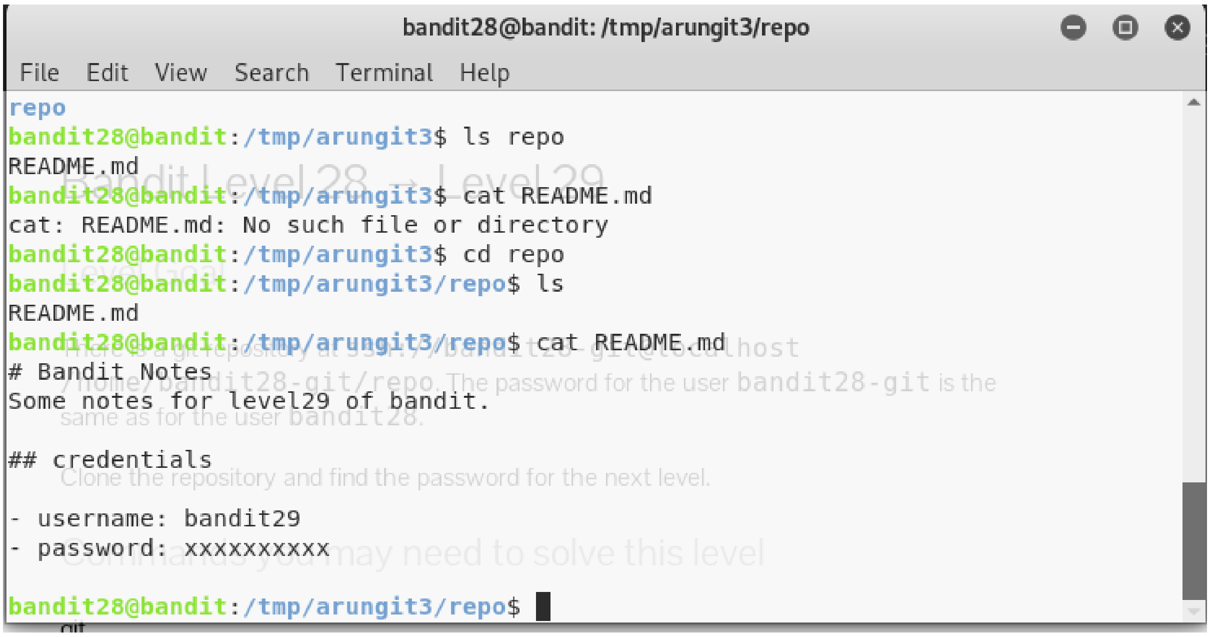
bandit28@bandit: /tmp/arungit3/repo
File Edit View Search Terminal Help

```
bandit28@bandit:~$ cd /tmp
bandit28@bandit:/tmp$ mkdir arungit3
bandit28@bandit:/tmp$ cd arungit3
bandit28@bandit:/tmp/arungit3$ ls
bandit28@bandit:/tmp/arungit3$ git clone ssh://bandit28-git@localhost/home/bandi
t28-git/repo
Cloning into 'repo'...
Could not create directory '/home/bandit28/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit28/.ssh/known_hos
ts).
This is a OverTheWire game server. More information on http://www.overthewire.or
g/wargames

bandit28-git@localhost's password:
Permission denied, please try again.
bandit28-git@localhost's password:
remote: Counting objects: 9, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 9 (delta 2), reused 0 (delta 0)
Receiving objects: 100% (9/9), done.
Resolving deltas: 100% (2/2), done.
bandit28@bandit:/tmp/arungit3$ ls
repo
```

After login in using bandit27 key, change the directory to /tmp and create a directory as I created arungit3 ,

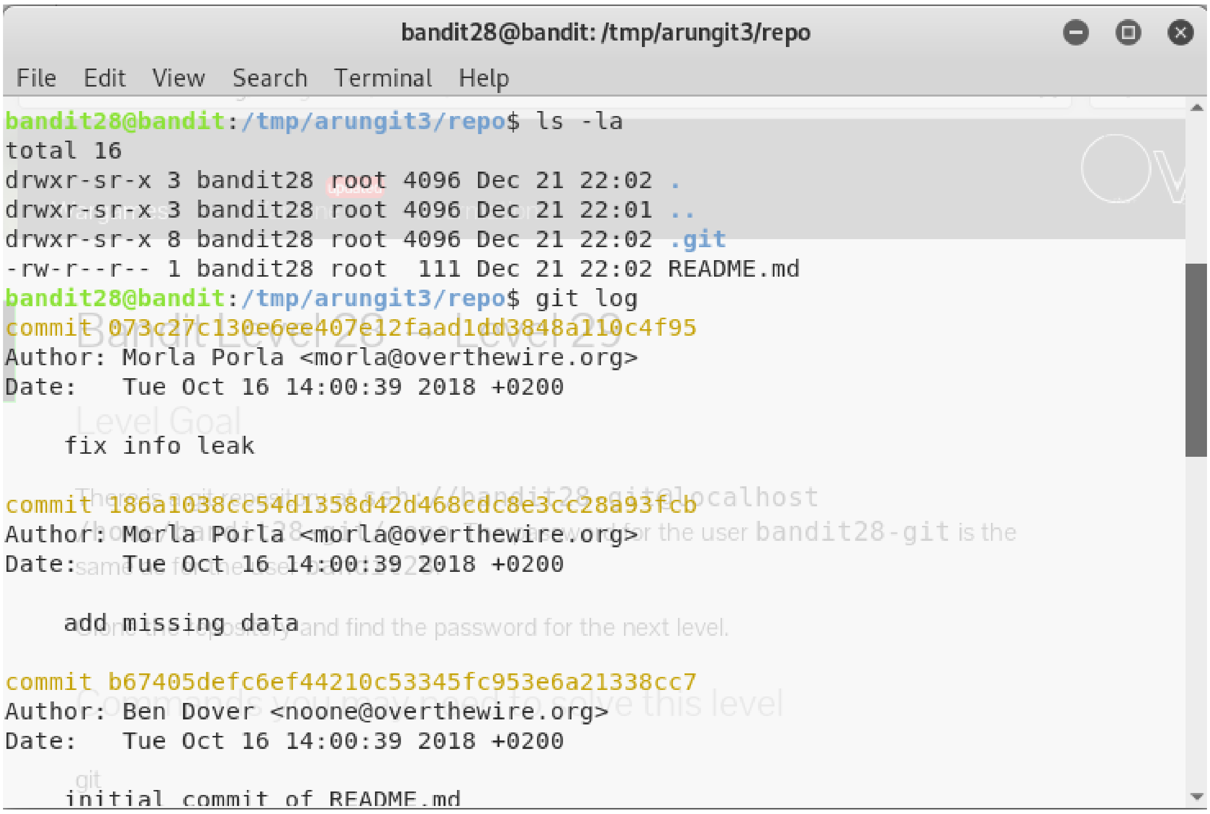
Using get clone, we receive an address to clone the repository through ssh.

We can check inside the files using ls command and it will display the repo directory.


bandit28@bandit: /tmp/arungit3/repo
File Edit View Search Terminal Help

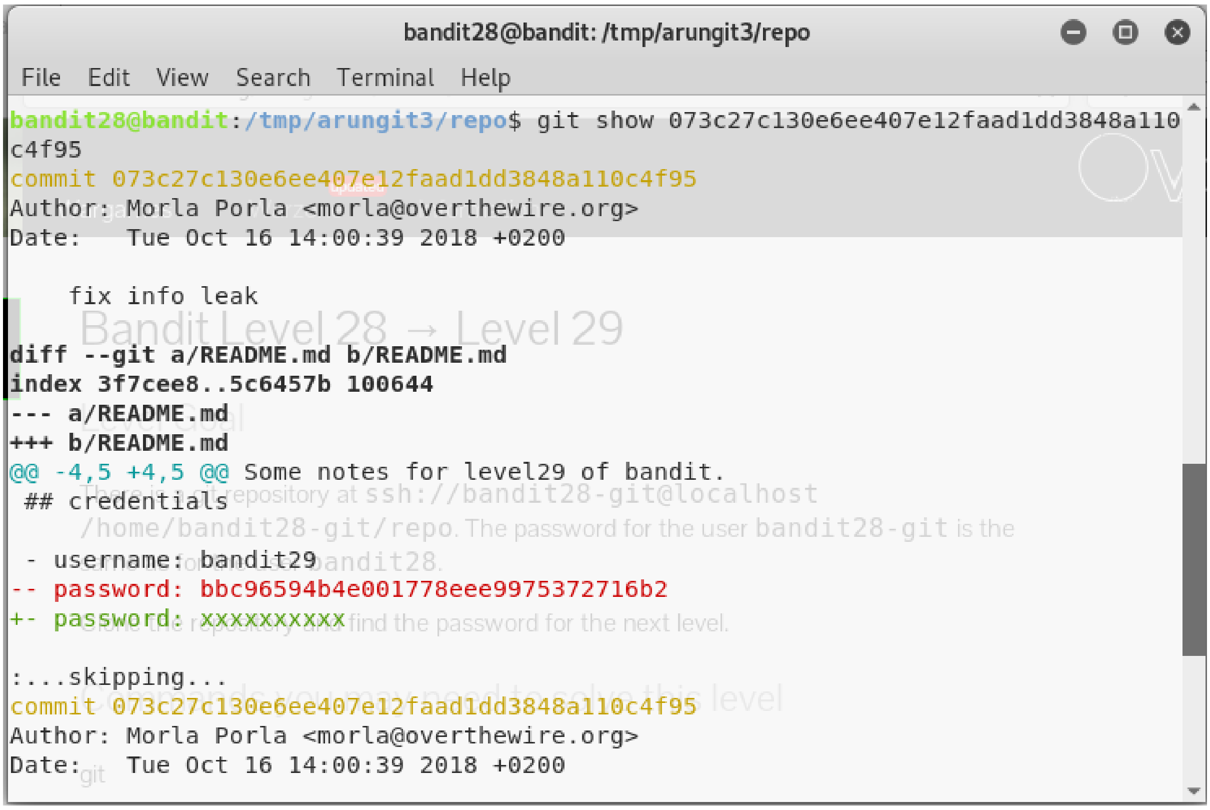
```
repo
bandit28@bandit:/tmp/arungit3$ ls repo
README.md
bandit28@bandit:/tmp/arungit3$ cat README.md
cat: README.md: No such file or directory
bandit28@bandit:/tmp/arungit3$ cd repo
bandit28@bandit:/tmp/arungit3/repo$ ls
README.md
bandit28@bandit:/tmp/arungit3/repo$ cat README.md
# Bandit Notes
Some notes for level29 of bandit.

## credentials

- username: bandit29
- password: xxxxxxxxxx

bandit28@bandit:/tmp/arungit3/repo$ 
```

After that , you can find the README.md file, Use the cat command to display the text, but it is showing like credentials .


bandit28@bandit: /tmp/arungit3/repo
File Edit View Search Terminal Help

```
bandit28@bandit:/tmp/arungit3/repo$ ls -la
total 16
drwxr-sr-x 3 bandit28 root 4096 Dec 21 22:02 .
drwxr-sr-x 3 bandit28 root 4096 Dec 21 22:01 ..
drwxr-sr-x 8 bandit28 root 4096 Dec 21 22:02 .git
-rw-r--r-- 1 bandit28 root  111 Dec 21 22:02 README.md
bandit28@bandit:/tmp/arungit3/repo$ git log
commit 073c27c130e6ee407e12faad1dd3848a110c4f95
Author: Morla Porla <morla@overthewire.org>
Date:   Tue Oct 16 14:00:39 2018 +0200

    fix info leak

commit 186a1038cc54d1358d42d468cdc8e3cc28a93fcb
Author: Morla Porla <morla@overthewire.org>
Date:   Tue Oct 16 14:00:39 2018 +0200

    add missing data

commit b67405defc6ef44210c53345fc953e6a21338cc7
Author: Ben Dover <noone@overthewire.org>
Date:   Tue Oct 16 14:00:39 2018 +0200

    initial commit of README.md
```

Use git log to get all passwords inside the git folder.


bandit28@bandit: /tmp/arungit3/repo
File Edit View Search Terminal Help

```
bandit28@bandit:/tmp/arungit3/repo$ git show 073c27c130e6ee407e12faad1dd3848a110
c4f95
commit 073c27c130e6ee407e12faad1dd3848a110c4f95
Author: Morla Porla <morla@overthewire.org>
Date:   Tue Oct 16 14:00:39 2018 +0200

    fix info leak

diff --git a/README.md b/README.md
index 3f7cee8..5c6457b 100644
--- a/README.md
+++ b/README.md
@@ -4,5 +4,5 @@ Some notes for level29 of bandit.
 ## credentials

 - username: bandit29
-- password: bbc96594b4e001778eee9975372716b2
+- password: xxxxxxxxxx

:...skipping...
commit 073c27c130e6ee407e12faad1dd3848a110c4f95
Author: Morla Porla <morla@overthewire.org>
Date:   Tue Oct 16 14:00:39 2018 +0200
```

Using git show command, we can get the password inside the author, this command will give you all credentials including password for the next level,

But you must check all passwords for entering the next level. After few attempts, I found the first one is correct.

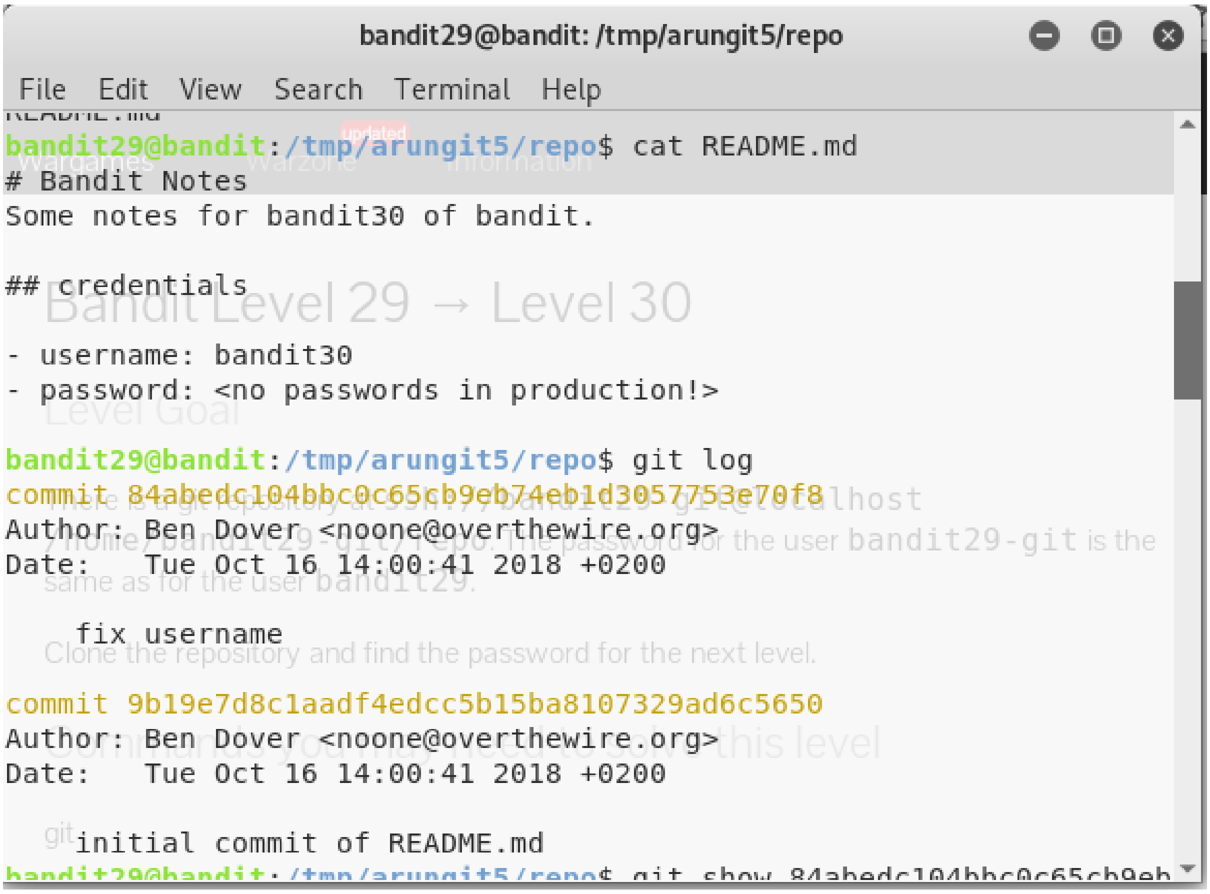
Password for the next level is **: bbc96594b4e001778eee9975372716b2**

Bandit Level 29 → Level 30

Level Goal

There is a git repository at ssh://bandit29-git@localhost/home/bandit29-git/repo. The password for the user bandit29-git is the same as for the user bandit29.

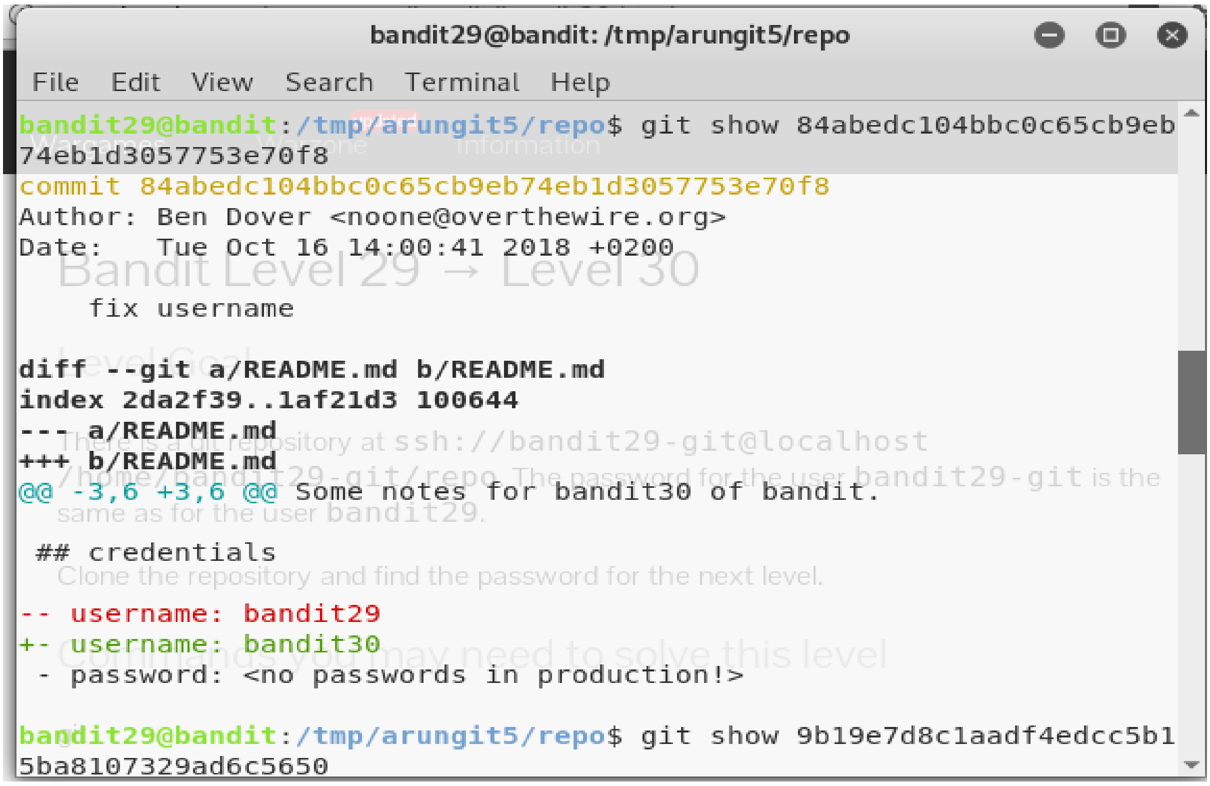
Clone the repository and find the password for the next level.


bandit29@bandit: /tmp/arungit5/repo
File Edit View Search Terminal Help

```
bandit29@bandit:/tmp/arungit5/repo$ cat README.md
# Bandit Notes
Some notes for bandit30 of bandit.

## credentials

- username: bandit30
- password: <no passwords in production!>

bandit29@bandit:/tmp/arungit5/repo$ git log
commit 84abedc104bbc0c65cb9eb74eb1d3057753e70f8
Author: Ben Dover <noone@overthewire.org>
Date:   Tue Oct 16 14:00:41 2018 +0200

    fix username

commit 9b19e7d8c1aadf4edcc5b15ba8107329ad6c5650
Author: Ben Dover <noone@overthewire.org>
Date:   Tue Oct 16 14:00:41 2018 +0200

    initial commit of README.md
bandit29@bandit:/tmp/arungit5/repo$ git show 84abedc104bbc0c65cb9eb
```

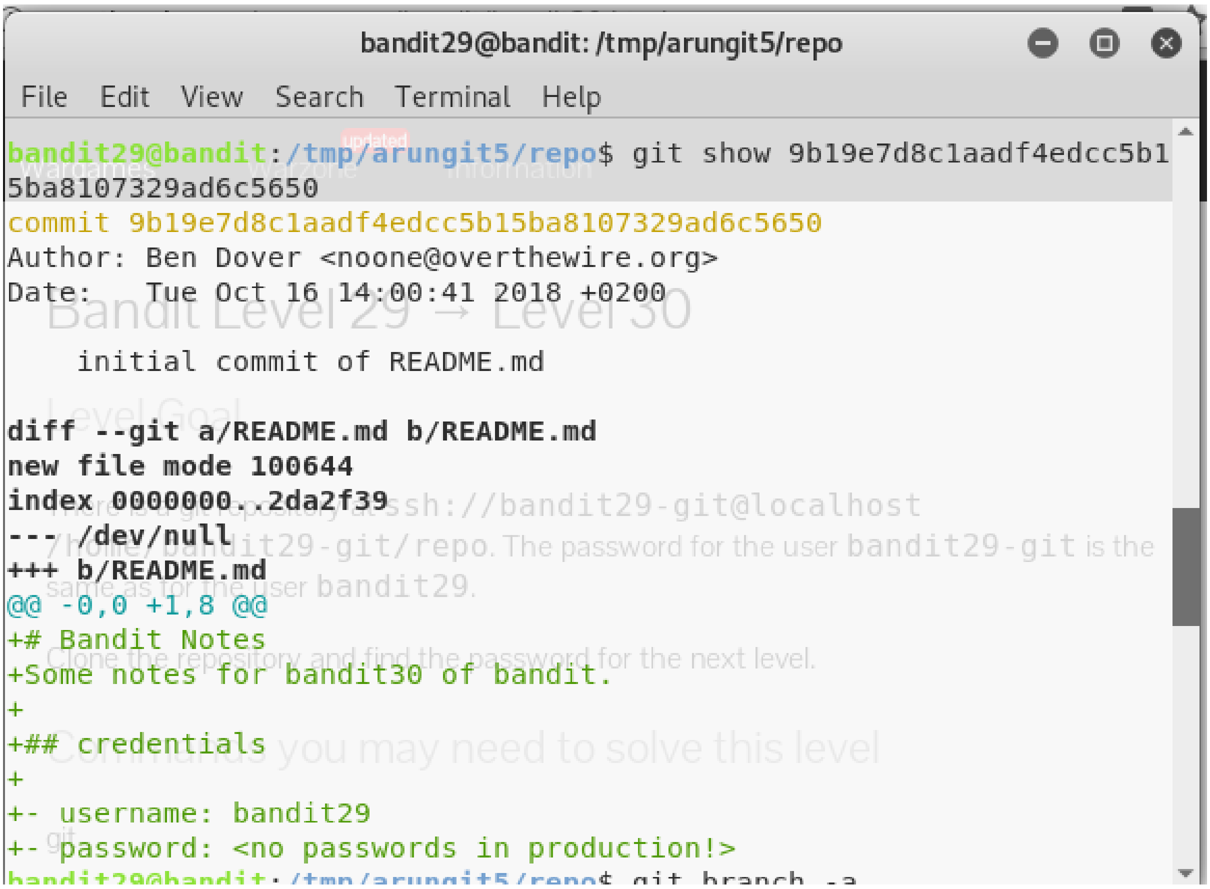
After using git clone command , we can check the README file inside arunfit5/repo file, here it is showing "<no passwords in production!>"

Using git log command , you can see two commit files with Author and Date


bandit29@bandit: /tmp/arungit5/repo
File Edit View Search Terminal Help

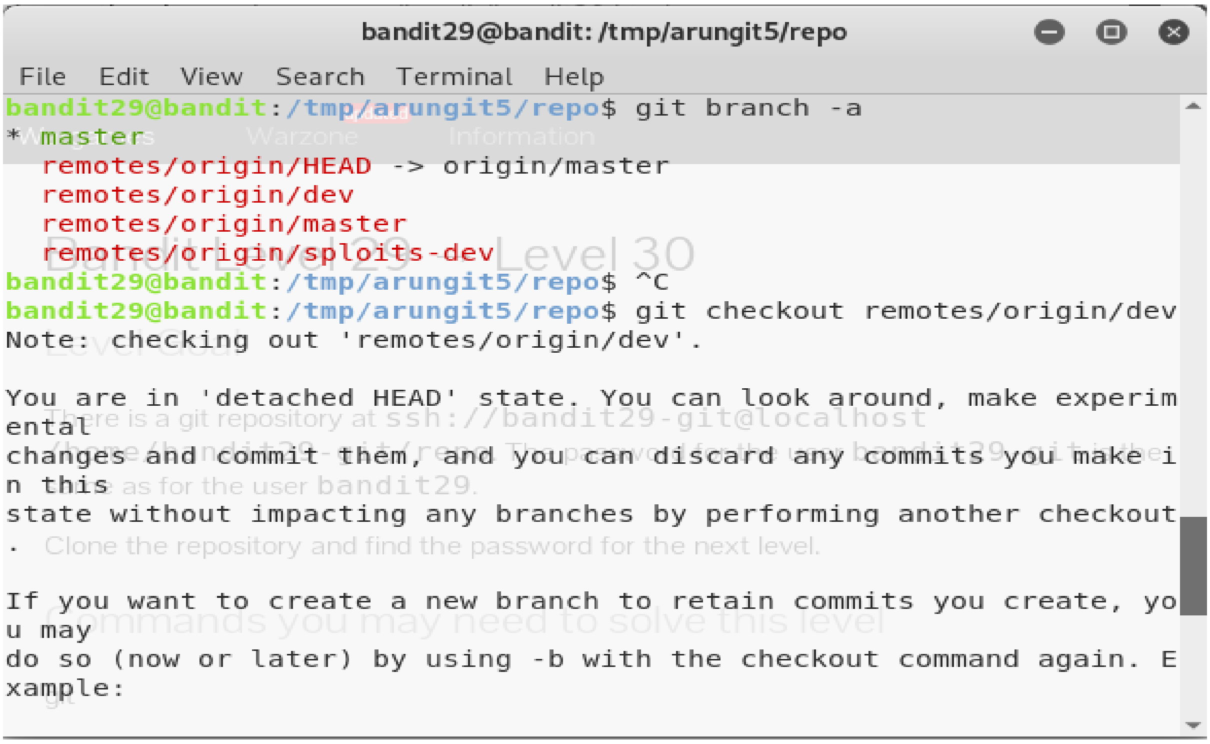
```
bandit29@bandit:/tmp/arungit5/repo$ git show 84abedc104bbc0c65cb9eb
74eb1d3057753e70f8
commit 84abedc104bbc0c65cb9eb74eb1d3057753e70f8
Author: Ben Dover <noone@overthewire.org>
Date:   Tue Oct 16 14:00:41 2018 +0200

    fix username

diff --git a/README.md b/README.md
index 2da2f39..1af21d3 100644
--- a/README.md
+++ b/README.md
@@ -3,6 +3,6 @@ Some notes for bandit30 of bandit.

 ## credentials

-- username: bandit29
+- username: bandit30
 - password: <no passwords in production!>

bandit29@bandit:/tmp/arungit5/repo$ git show 9b19e7d8c1aadf4edcc5b1
5ba8107329ad6c5650
```

Using git show command with the commit , you can check the first password and it is showing same as first one.


bandit29@bandit: /tmp/arungit5/repo
File Edit View Search Terminal Help

```
bandit29@bandit:/tmp/arungit5/repo$ git show 9b19e7d8c1aadf4edcc5b1
5ba8107329ad6c5650
commit 9b19e7d8c1aadf4edcc5b15ba8107329ad6c5650
Author: Ben Dover <noone@overthewire.org>
Date:   Tue Oct 16 14:00:41 2018 +0200

    initial commit of README.md

diff --git a/README.md b/README.md
new file mode 100644
index 0000000..2da2f39
--- /dev/null
+++ b/README.md
@@ -0,0 +1,8 @@
+# Bandit Notes
+Some notes for bandit30 of bandit.
+
+## credentials
+
+- username: bandit29
+- password: <no passwords in production!>
bandit29@bandit:/tmp/arungit5/repo$ git branch -a
```

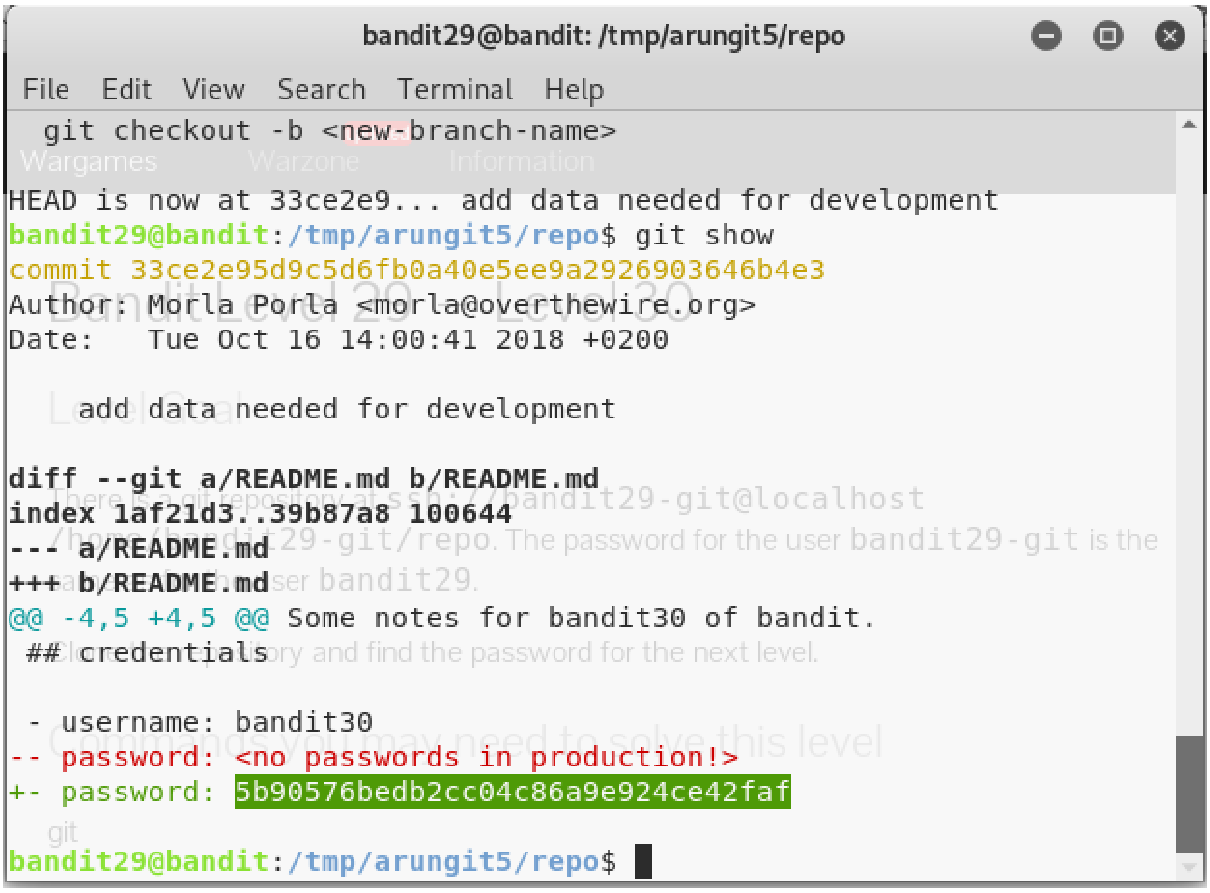
Overthegame (Bandit)


bandit29@bandit: /tmp/arungit5/repo
File Edit View Search Terminal Help

```
bandit29@bandit:/tmp/arungit5/repo$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/dev
  remotes/origin/master
  remotes/origin/sploits-dev
bandit29@bandit:/tmp/arungit5/repo$ ^C
bandit29@bandit:/tmp/arungit5/repo$ git checkout remotes/origin/dev
Note: checking out 'remotes/origin/dev'.

You are in 'detached HEAD' state. You can look around, make experim
ental
changes and commit them, and you can discard any commits you make i
n this
state without impacting any branches by performing another checkout
.

If you want to create a new branch to retain commits you create, yo
u may
do so (now or later) by using -b with the checkout command again. E
xample:
```

Using git branch command, you can see the files inside the git folder files, it contains origin/master file and other files,

You can check the remotes/origin/dev file,


bandit29@bandit: /tmp/arungit5/repo
File Edit View Search Terminal Help

```
  git checkout -b <new-branch-name>

HEAD is now at 33ce2e9... add data needed for development
bandit29@bandit:/tmp/arungit5/repo$ git show
commit 33ce2e95d9c5d6fb0a40e5ee9a2926903646b4e3
Author: Morla Porla <morla@overthewire.org>
Date:   Tue Oct 16 14:00:41 2018 +0200

    add data needed for development

diff --git a/README.md b/README.md
index 1af21d3..39b87a8 100644
--- a/README.md
+++ b/README.md
@@ -4,5 +4,5 @@ Some notes for bandit30 of bandit.
 ## credentials

 - username: bandit30
-- password: <no passwords in production!>
+- password: 5b90576bedb2cc04c86a9e924ce42faf

bandit29@bandit:/tmp/arungit5/repo$
```

Again, use git show command to check inside the data it contains Author and Date of modification, finally we found the password for next level.

Next level password is : **5b90576bedb2cc04c86a9e924ce42faf**

I completed the Level 30 and next levels are same as level 29 and 30 using got commands.

Thank you for sharing the website and this web site is very useful to me.

In my personal opinion I can say that, I learned a lot of new commands in Linux.