OverTheWire Bandit

Levels 1-34 Walkthrough

Original Author(s): *Pavandeep Singh*

Table of Contents

Abstract

This report will provide a walkthrough of a popular wargame called *Bandit*. The main objective of this challenge is to teach the basics of Linux commands in an engaging and practical way. The game consists of 34 levels, each designed to introduce new concepts and commands step by step.

This wargame, hosted by the *OverTheWire* organization, is aimed at absolute beginners who want to strengthen their Linux skills while solving real challenges. The lab is not difficult if you have the right basic knowledge and a willingness to experiment.

Disclaimer: This report is provided for educational and informational purpose only (Penetration Testing). Penetration Testing refers to legal intrusion tests that aim to identify vulnerabilities and improve cybersecurity, rather than for malicious purposes.

Level 0

This is a pretty simple level. It teaches us to connect to a host using SSH. This is going to teach players the usage of SSH command.

We got the required information from reading the instruction page.

Host: bandit.labs.overthewire.org

Port: 2220

Username: bandit0

Password: bandit0

We used the above information to login using ssh as shown in the given image.

```
ssh bandit0@bandit.labs.overthewire.org -p 2220
```

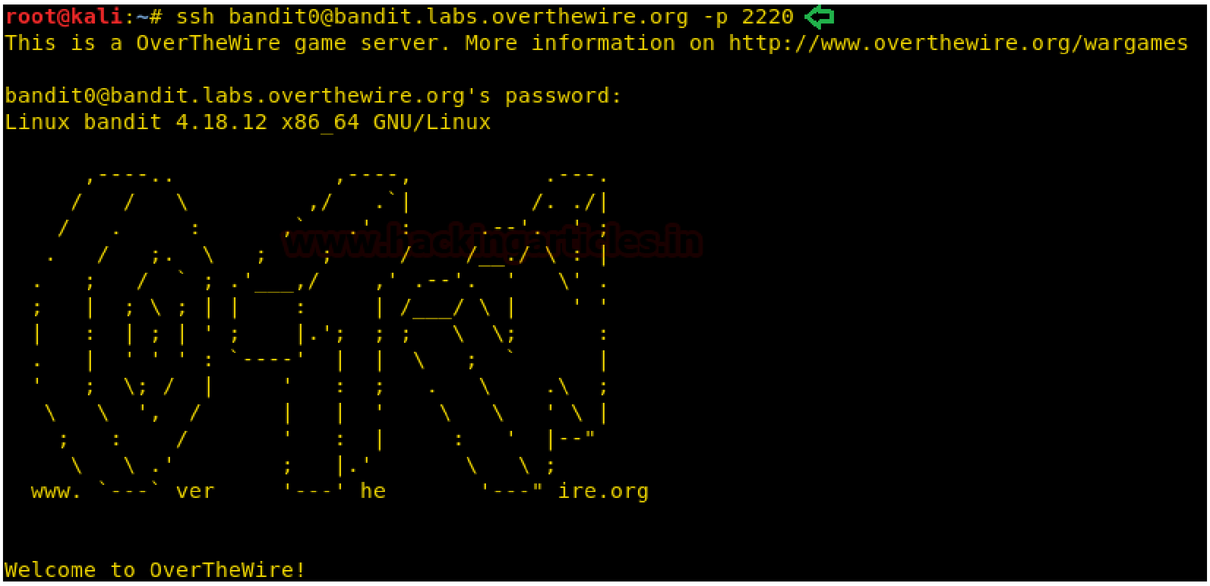
This level doesn't require anything else other than logging in. Time to move in on the next level.

Level 0-1

Now, from the bandit0 shell, we need to find the password for logging as the next user. To find that password, we are going to list files in the directory. Our target is to find a file named readme. After finding that file, we need to read the password stored inside that file.

We use the ls command to list the files in the current directory. We found the readme file. Now to read the password we will use the cat command. After that, we are going to use the password to login into next level using SSH

```
ls -la
cat readme
ssh bandit1@localhost
```

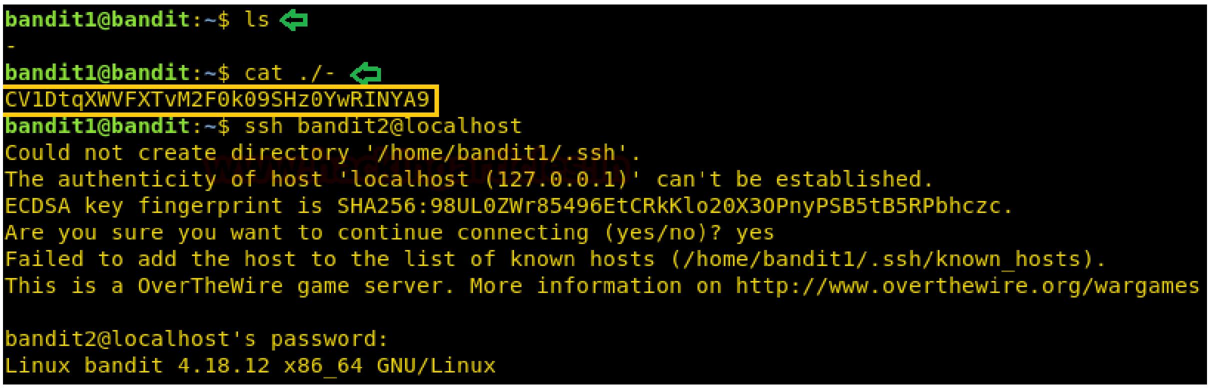


```
root@kali:~# ssh bandit0@bandit.labs.overthewire.org -p 2220
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

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

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


Welcome to OverTheWire!
```

Level 1-2

We are informed that the password for the next level is stored inside a file named -(hyphen). So, to find it we use the ls command. Now comes the part where we have to read the file. As the file is named -(hyphen) we won't be able to read it simply by cat command. As cat command considers -(hyphen) as stdin/Stout. If we directly use cat command, it won't be able to understand that hyphen is a file name. So, we will prefix the command with the path ./, This will help us to read the password stored as shown in the given figure. Since we found the password for the user bandit2. We will use it to get an SSH connection as bandit2.

```
ls
cat ./-
ssh bandit2@localhost
```

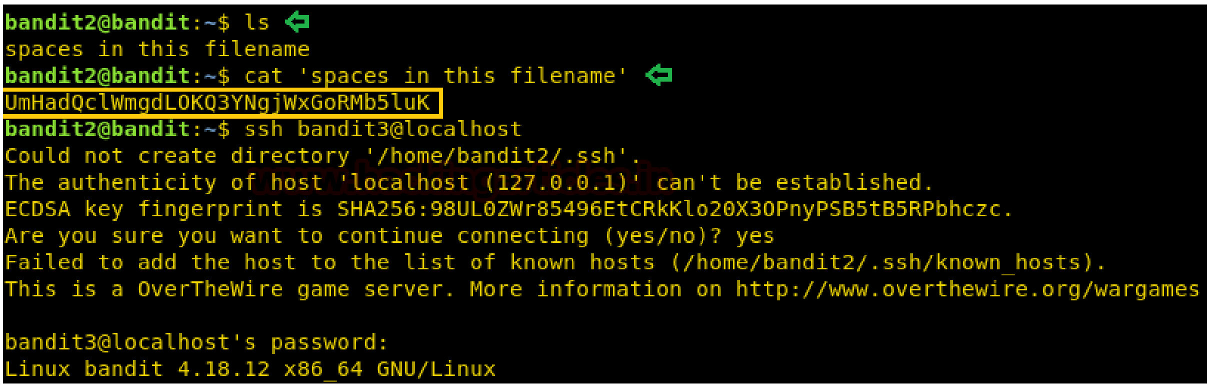


```
bandit1@bandit:~$ ls
-
bandit1@bandit:~$ cat ./-
CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9
bandit1@bandit:~$ ssh bandit2@localhost
Could not create directory '/home/bandit1/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit1/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit2@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 2-3

We are informed that the password for the next level is stored inside a file named spaces in this filename. So, to find it we use the ls command. Now comes the part where we have to read the file. As the file is named spaces in this filename, we won't be able to read it simply by cat command. As cat command reads files name only until space as it considers space as null '/0'. If we directly use cat command, it won't be able to find the file. So, we will write the name of the file in quotes, this will help us to read the password stored as shown in the given figure. Since we found the password for the user bandit3. We will use it to get an SSH connection as bandit3.

```
ls
cat 'spaces in this filename'
ssh bandit3@localhost
```

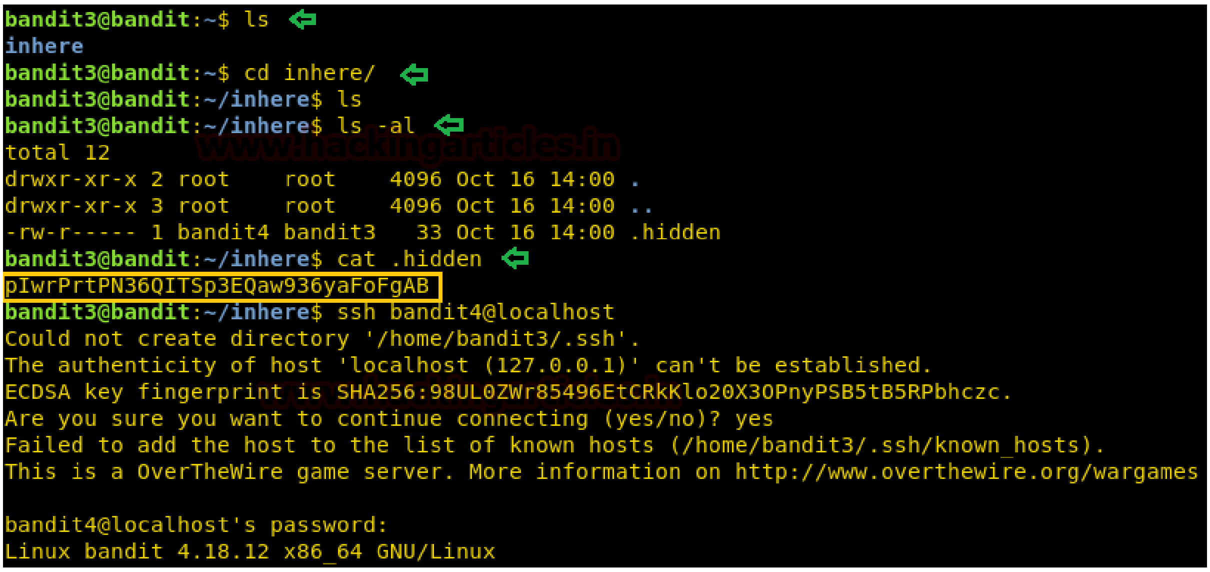


```
bandit2@bandit:~$ ls
spaces in this filename
bandit2@bandit:~$ cat 'spaces in this filename'
UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK
bandit2@bandit:~$ ssh bandit3@localhost
Could not create directory '/home/bandit2/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit2/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit3@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 3-4

We are informed that the password for the next level is stored inside a directory named inhere. So, to find it we use the ls command. Now, after traversing inside inhere directory we run ls command again. Now it might be the case that the file is hidden. So, we run ls command with -al parameter. It lists all files including the hidden one. And we found the .hidden file. In Linux, the file with a dot(.) in front of the name of the file makes it hidden. Now we would simply use the cat command to read the password stored in the file. Since we found the password for the user bandit4. We will use it to get an SSH connection as bandit4.

```
ls
cd inhere/
ls
ls -al
cat .hidden
ssh bandit4@localhost
```

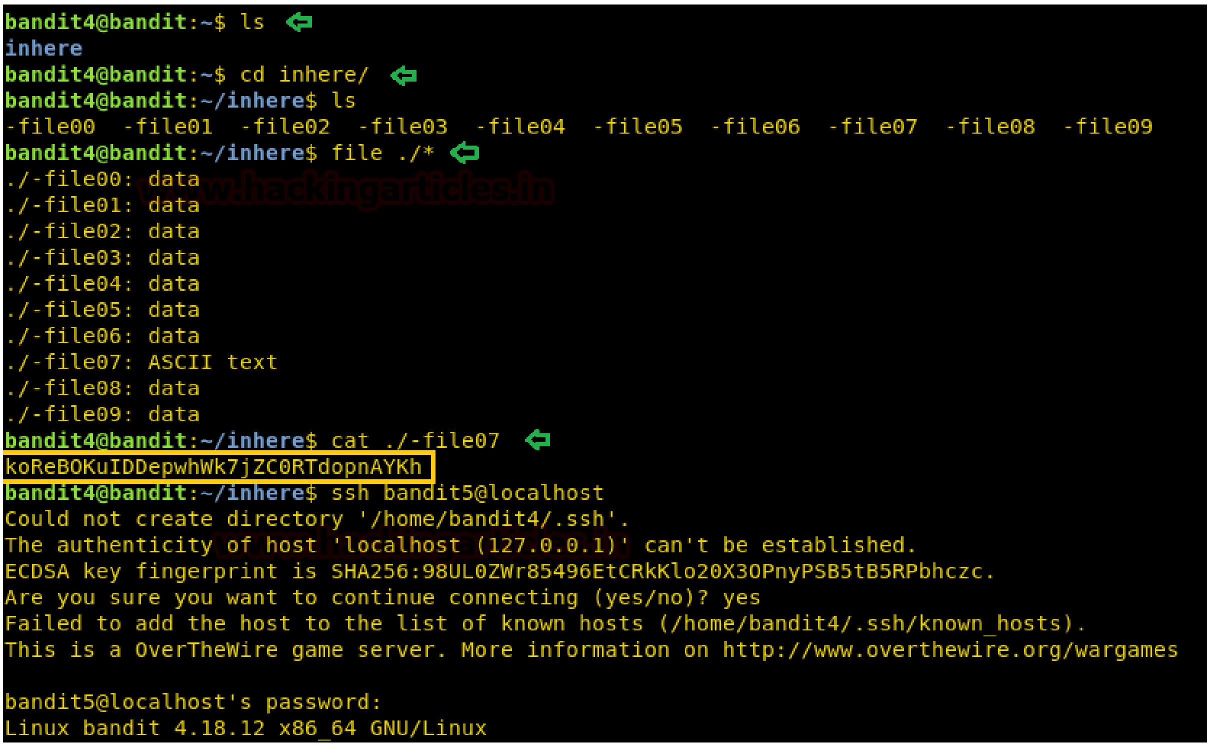


```
bandit3@bandit:~$ ls
inhere
bandit3@bandit:~$ cd inhere/
bandit3@bandit:~/inhere$ ls
bandit3@bandit:~/inhere$ ls -al
total 12
drwxr-xr-x 2 root     root     4096 Oct 16 14:00 .
drwxr-xr-x 3 root     root     4096 Oct 16 14:00 ..
-rw-r----- 1 bandit4 bandit3   33 Oct 16 14:00 .hidden
bandit3@bandit:~/inhere$ cat .hidden
pIwrPrtPN36QITSp3EQaw936yaFoFgAB
bandit3@bandit:~/inhere$ ssh bandit4@localhost
Could not create directory '/home/bandit3/.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/bandit3/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit4@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 4-5

We are informed that the password for the next level is stored inside a human-readable file. So, to find it we use the ls command. Now, after traversing inside inhere directory we run ls command again. This gives us a bunch of files as shown in the image. We will use the file command to get the information about the files. From files command, we now know that the file07 contains ASCII text. It is mostly readable text. So, let's read it using cat command. This gives us the password for the next level. We will use it to get an SSH connection as bandit5.

```
ls -la
cd inhere/
ls
file ./*
cat ./-file07
ssh bandit5@localhost
```

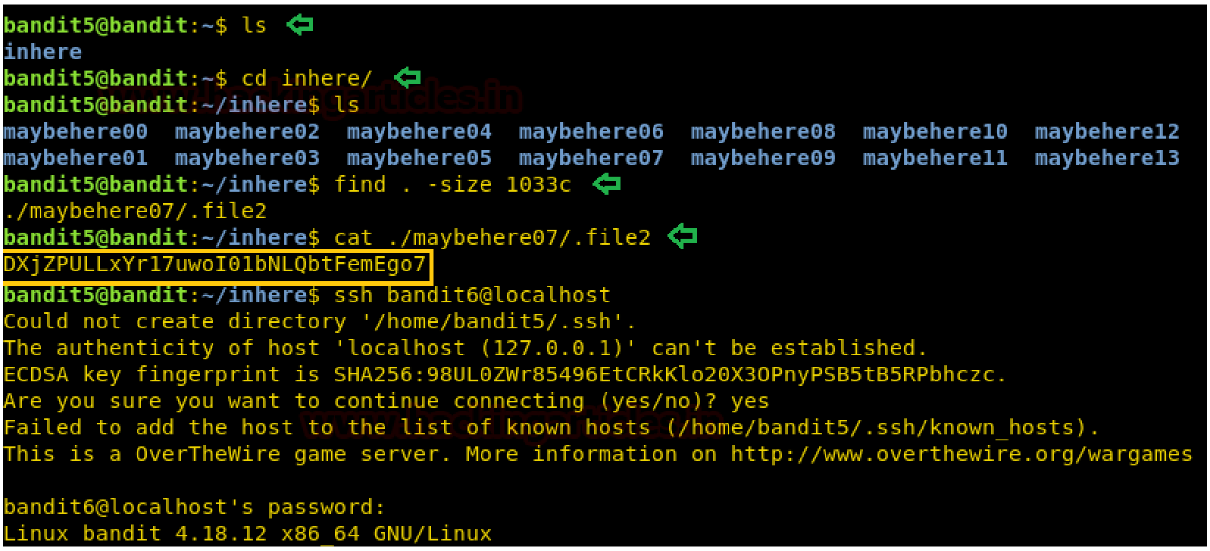


```
bandit4@bandit:~$ ls
inhere
bandit4@bandit:~$ cd inhere/
bandit4@bandit:~/inhere$ ls
-file00  -file01  -file02  -file03  -file04  -file05  -file06  -file07  -file08  -file09
bandit4@bandit:~/inhere$ file ./*
./-file00: data
./-file01: data
./-file02: data
./-file03: data
./-file04: data
./-file05: data
./-file06: data
./-file07: ASCII text
./-file08: data
./-file09: data
bandit4@bandit:~/inhere$ cat ./-file07
koReBOKuIDDepwhWk7jZC0RTdopnAYKh
bandit4@bandit:~/inhere$ ssh bandit5@localhost
Could not create directory '/home/bandit4/.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/bandit4/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit5@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 5-6

We are informed that the password for the next level is stored inside a directory named inhere. So, to find it we use the ls command. Now, after traversing inside inhere directory we run ls command again. This gives us a bunch of files as shown in the image. We will use the file size to find the file. Find command has the parameter of size in which we have to use 'c' for depicting size in bytes. From find command, we now know that the file2 contains the password. So, let's read it using cat command. This gives us the password for the next level. We will use it to get an SSH connection as bandit6.

```
ls
cd inhere/
ls
find . -size 1033c
cat ./maybehere07/.file2
ssh bandit6@localhost
```

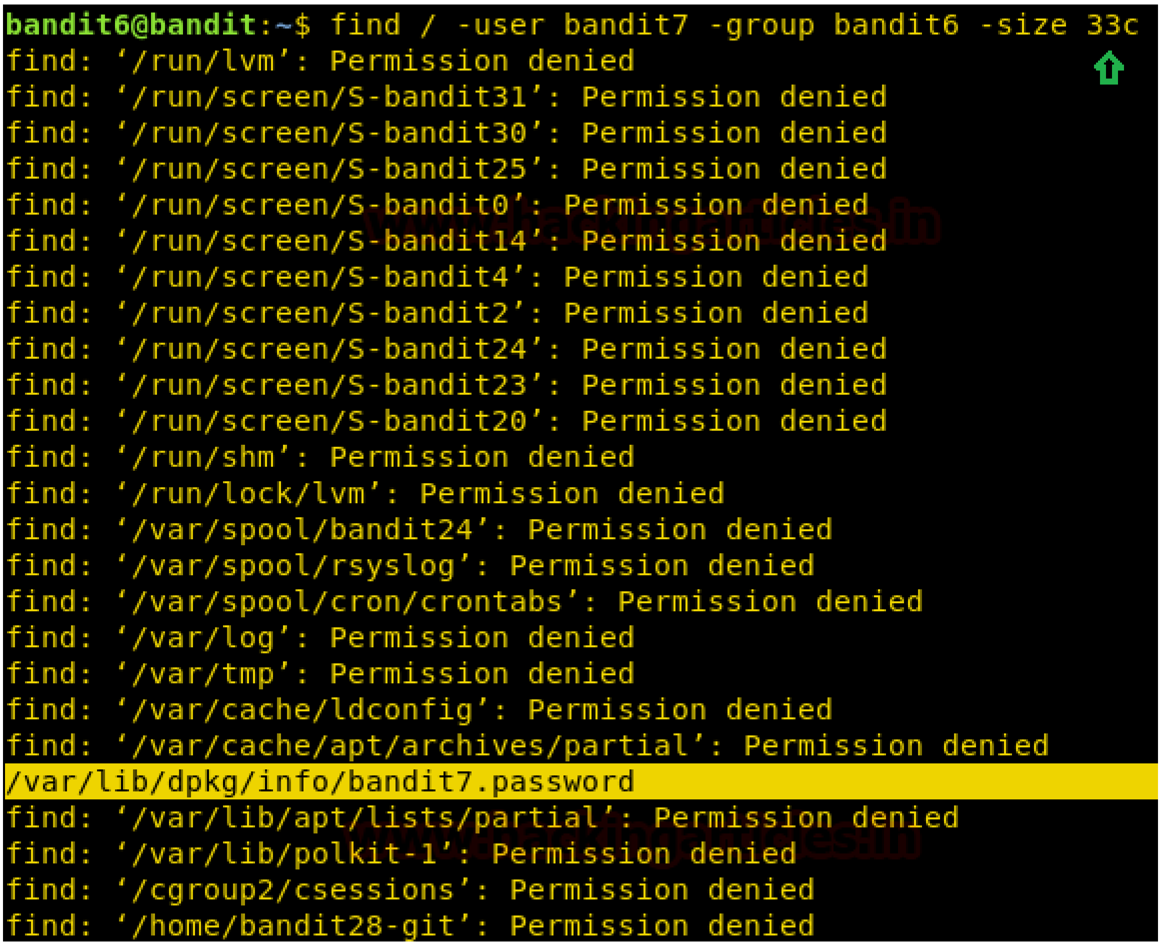


```
bandit5@bandit:~$ ls
inhere
bandit5@bandit:~$ cd inhere/
bandit5@bandit:~/inhere$ ls
maybehere00  maybehere02  maybehere04  maybehere06  maybehere08  maybehere10  maybehere12
maybehere01  maybehere03  maybehere05  maybehere07  maybehere09  maybehere11  maybehere13
bandit5@bandit:~/inhere$ find . -size 1033c
./maybehere07/.file2
bandit5@bandit:~/inhere$ cat ./maybehere07/.file2
DXjZPULLxYr17uwoI01bNLQbtFemEgo7
bandit5@bandit:~/inhere$ ssh bandit6@localhost
Could not create directory '/home/bandit5/.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/bandit5/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit6@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 6-7

We are informed that the password for the next level is stored somewhere on the server. So, finding the file over the server would be a lot trickier if we are using ls. So, we will try to widen our scope of search using the find command. We are hinted that the user of the file is bandit7 and it is a part of group bandit 6. We will add this information as parameters in the find command. We are given the size too. Let’s add that too. Now as we can see in the given image, we successfully located the password file hidden over the server.

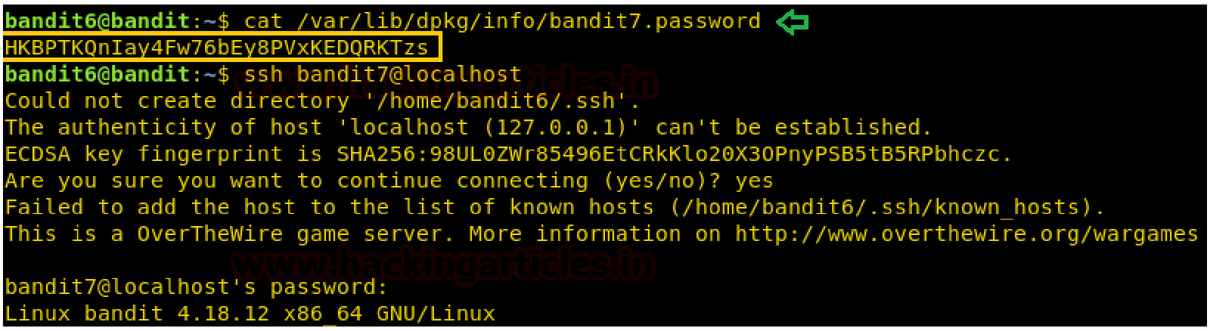
```
find / -user bandit7 -group bandit6 -size 33c
```



```
bandit6@bandit:~$ find / -user bandit7 -group bandit6 -size 33c
find: '/run/lvm': Permission denied
find: '/run/screen/S-bandit31': Permission denied
find: '/run/screen/S-bandit30': Permission denied
find: '/run/screen/S-bandit25': Permission denied
find: '/run/screen/S-bandit0': Permission denied
find: '/run/screen/S-bandit14': Permission denied
find: '/run/screen/S-bandit4': Permission denied
find: '/run/screen/S-bandit2': Permission denied
find: '/run/screen/S-bandit24': Permission denied
find: '/run/screen/S-bandit23': Permission denied
find: '/run/screen/S-bandit20': Permission denied
find: '/run/shm': Permission denied
find: '/run/lock/lvm': Permission denied
find: '/var/spool/bandit24': 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/dpkg/info/bandit7.password
find: '/var/lib/apt/lists/partial': Permission denied
find: '/var/lib/polkit-1': Permission denied
find: '/cgroup2/csessions': Permission denied
find: '/home/bandit28-git': Permission denied
```

```
cat /var/lib/dpkg/info/bandit7.password

ssh bandit7@localhost
```

From find command, we now know that the bandit7.password contains the credentials. So, let's read it using cat command. This gives us the password for the next level. We will use it to get an SSH connection as bandit7.

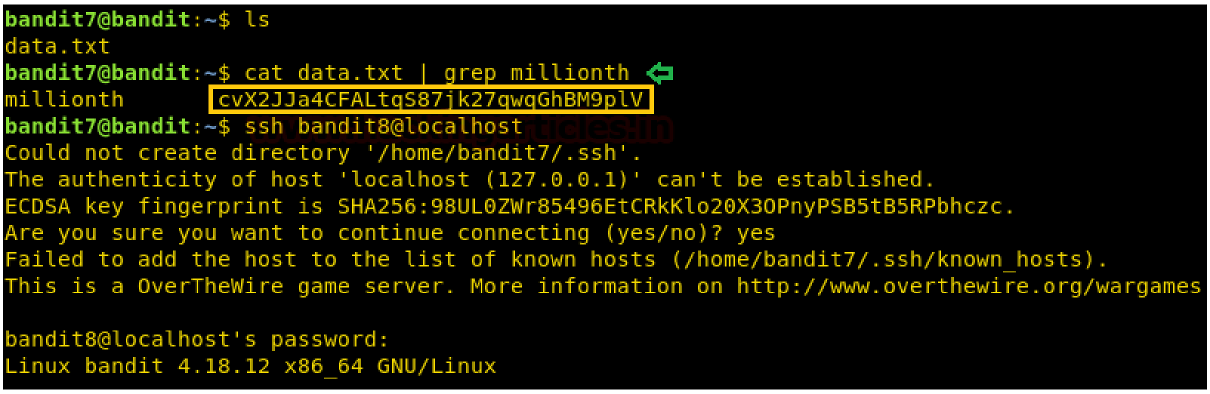


```
bandit6@bandit:~$ cat /var/lib/dpkg/info/bandit7.password
HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs
bandit6@bandit:~$ ssh bandit7@localhost
Could not create directory '/home/bandit6/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit6/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit7@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 7-8

We are informed that the password for the next level is stored inside a file named data.txt. So, to find it we use the ls command. Now we are hinted that the password is written next to the word millionth in the data.txt file. This means if we find the millionth word, we find the password. We are going to use the grep command for finding millionth. Here we using the (|) Unix pipe. The Pipe connects the standard output from the first command and feeds it as standard input to the second command. In our case, first cat command reads the file and then the data inside the file is sent to grep command to work on. This gives us the password for the next level. We will use it to get an SSH connection as bandit8.

```
ls
cat data.txt | grep millionth
ssh bandit8@localhost
```

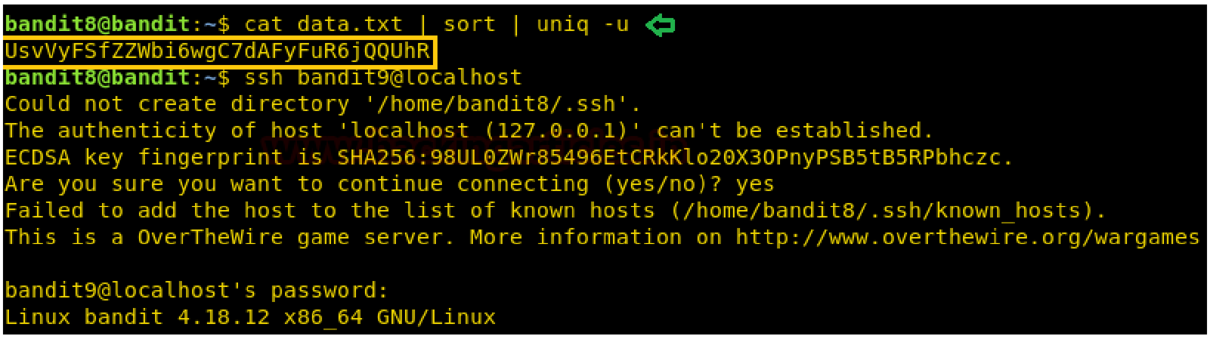


```
bandit7@bandit:~$ ls
data.txt
bandit7@bandit:~$ cat data.txt | grep millionth
millionth       cvX2JJa4CFALtqS87jk27qwqGhBM9plV
bandit7@bandit:~$ ssh bandit8@localhost
Could not create directory '/home/bandit7/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key 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/bandit7/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit8@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 8-9

We are informed that the password for the next level is stored inside a file named data.txt. It is hinted that the password is the only line of text that occurs only once. Here we are going to use sort command to sort the text inside the data.txt file. But still, the file contains a lot of repeating statements so we will use the uniq command to print the not repeating statement. We are using multiple pipes here to get a filtered result. This gives us the password for the next level. We will use it to get an SSH connection as bandit9.

```
cat data.txt | sort | uniq -u

ssh bandit9@localhost
```

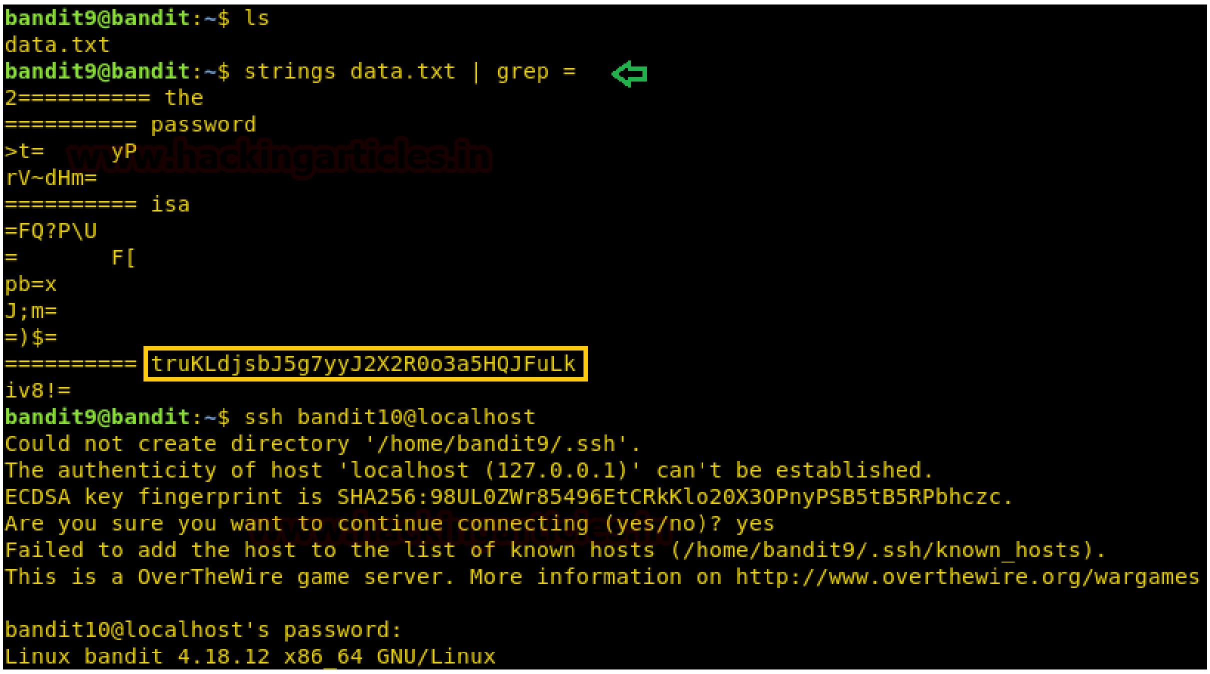


```
bandit8@bandit:~$ cat data.txt | sort | uniq -u
UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR
bandit8@bandit:~$ ssh bandit9@localhost
Could not create directory '/home/bandit8/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit8/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit9@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 9-10

We are informed that the password for the next level is stored inside a file named data.txt. We are hinted that the password is followed by several '=' characters. Now if we are to use the cat command our screen would be filled with unreadable mesh. So, to get a more refined approach we are going to use strings command which prints character sequences that are at least 4 characters long. And to get to the exact location of the password, we are going to use grep. This gives us the password for the next level. We will use it to get an SSH connection as bandit10.

```
ls

strings data.txt | grep =

ssh bandit10@localhost
```

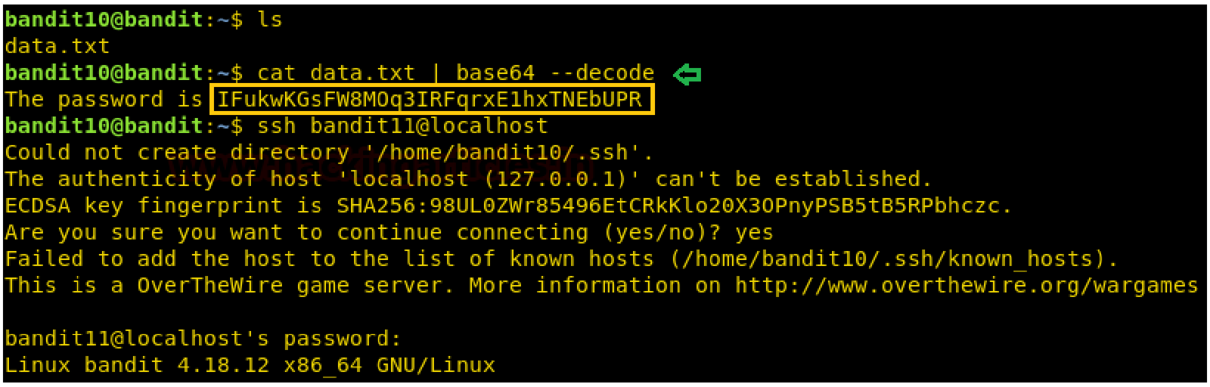


```
bandit9@bandit:~$ ls
data.txt
bandit9@bandit:~$ strings data.txt | grep =
2========== the
========== password
>t=     yP
rV~dHm=
========== isa
=FQ?P\U
=       F[
pb=x
J;m=
=)$=
========== truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk
iv8!=
bandit9@bandit:~$ ssh bandit10@localhost
Could not create directory '/home/bandit9/.ssh'.
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/bandit9/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit10@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 10-11

We are informed that the password for the next level is stored inside a file named data.txt. So, to find it we use the ls command. Now, we are hinted that the password is encrypted in Base64. Now we can either read the file with cat command and decode the Base64 manually but we have a command in Linux that can do the heavy lifting for us. So, we use piping to use cat command and base64 command with d parameter to read and decode the text simultaneously. This gives us the password for the next level. We will use it to get an SSH connection as bandit11.

```
ls
cat data.txt | base64 --decode
ssh bandit11@localhost
```

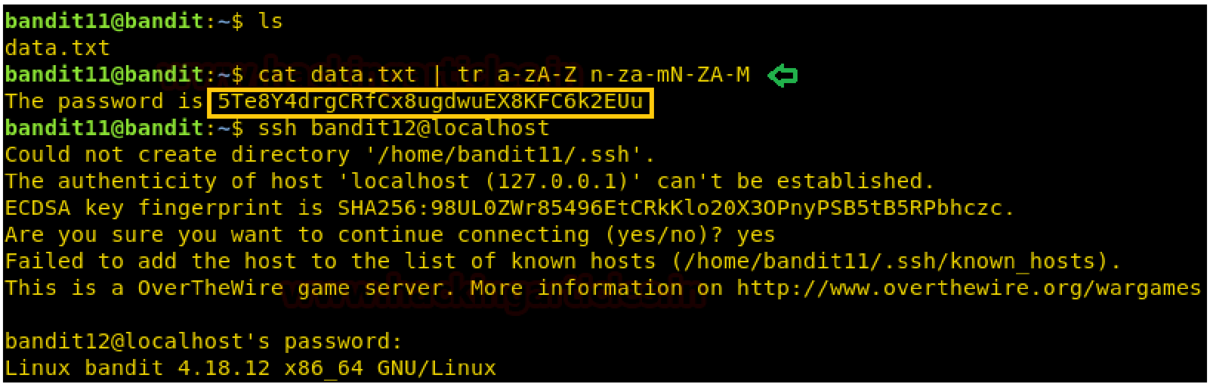


```
bandit10@bandit:~$ ls
data.txt
bandit10@bandit:~$ cat data.txt | base64 --decode
The password is IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR
bandit10@bandit:~$ ssh bandit11@localhost
Could not create directory '/home/bandit10/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA 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/bandit10/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit11@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 11-12

We are informed that the password for the next level is stored inside a file named data.txt. So, to find it we use the ls command. Now, we are hinted that the file containing the password has changed the format of letters in such a way that all the lowercase and uppercase letters have been rotated by 13 positions. If we can remember right that exactly what happens in ROT13 encryption. Now, to convert the text, we can use the 'tr' command. This command translates characters depending on the parameters provided. We used n-z and a-m because tr won't continue to translate after the Z. This gives us the password for the next level. We will use it to get an SSH connection as bandit12.

```
ls
cat data.txt | tr a-zA-Z n-za-mN-ZA-M
ssh bandit12@localhost
```

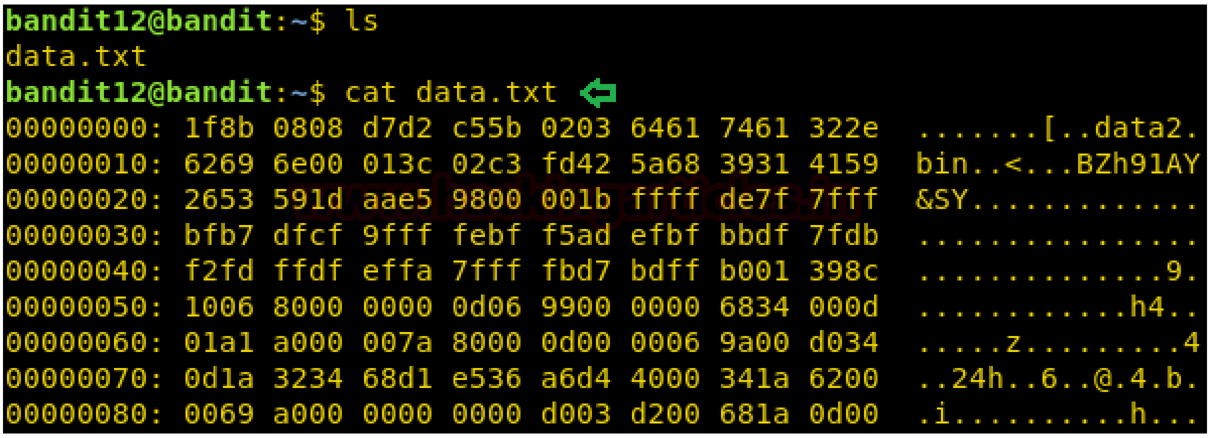


```
bandit11@bandit:~$ ls
data.txt
bandit11@bandit:~$ cat data.txt | tr a-zA-Z n-za-mN-ZA-M
The password is 5Te8Y4drgCRfCx8ugdwuEX8KFC6k2EUu
bandit11@bandit:~$ ssh bandit12@localhost
Could not create directory '/home/bandit11/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
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/bandit11/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit12@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 12-13

We are informed that the password for the next level is stored inside a directory named inhere. So, to find it we use the ls command. We are hinted that the file containing the password is in the form of a hex dump. Just out of curiosity, let's read the file using the cat command. As we can see in the given image that the password is not at all readable. We are also told that the password file has been repeatedly compressed. Now to decompress we are going to need a directory with read and write permissions. The tmp directory in root contains the required permissions.

```
ls
cat data.txt
```



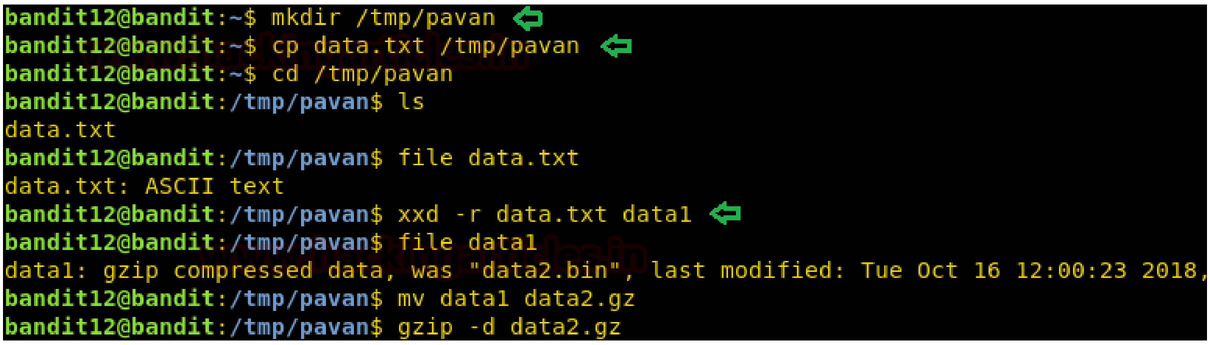
```
bandit12@bandit:~$ ls
data.txt
bandit12@bandit:~$ cat data.txt
00000000: 1f8b 0808 d7d2 c55b 0203 6461 7461 322e  .......[..data2.
00000010: 6269 6e00 013c 02c3 fd42 5a68 3931 4159  bin..<...BZh91AY
00000020: 2653 591d aae5 9800 001b ffff de7f 7fff  &SY.............
00000030: bfb7 dfcf 9fff febf f5ad efbf bbdf 7fdb  ................
00000040: f2fd ffdf effa 7fff fbd7 bdff b001 398c  ..............9.
00000050: 1006 8000 0000 0d06 9900 0000 6834 000d  ............h4..
00000060: 01a1 a000 007a 8000 0d00 0006 9a00 d034  .....z.........4
00000070: 0d1a 3234 68d1 e536 a6d4 4000 341a 6200  ..24h..6..@.4.b.
00000080: 0069 a000 0000 0000 d003 d200 681a 0d00  .i..........h...
```

So, let's create a directory inside the tmp directory. Here we named it pavan. Now for further operations let's copy the file in the directory we just created. Now let's traverse to our directory using the cd command. Now we check if we have our file in this directory. Now to understand the type of file we are going to use the file command it returns us the type of file. On running the command, we are informed that the file is ASCII text. But as we saw earlier that it is not readable. The xxd command is used in Linux to make the hexdump of a file. It is also used to reverse this process. Let's use it to retrieve the original file. We are going to use the 'r' parameter to revert the process and provide it with a filename where it should store its output. Here we will name it data1

Now it's time to check the retrieved file, we use the file command again. This tells us that it is a gzip compressed file.

Now decompress first, we need to rename the file and provide it with a proper gzip extension. We are going to use the move command for this. We renamed the file as data2.gz. Now using the gzip command and -d parameter, we decompress the file.

```
mkdir /tmp/pavan
cp data.txt /tmp/pavan
cd /tmp/pavan
ls
file data.txt
xxd -r data.txt data1
file data1
mv data1 data2.gz
gzip -d data2.gz
```



```
bandit12@bandit:~$ mkdir /tmp/pavan
bandit12@bandit:~$ cp data.txt /tmp/pavan
bandit12@bandit:~$ cd /tmp/pavan
bandit12@bandit:/tmp/pavan$ ls
data.txt
bandit12@bandit:/tmp/pavan$ file data.txt
data.txt: ASCII text
bandit12@bandit:/tmp/pavan$ xxd -r data.txt data1
bandit12@bandit:/tmp/pavan$ file data1
data1: gzip compressed data, was "data2.bin", last modified: Tue Oct 16 12:00:23 2018,
bandit12@bandit:/tmp/pavan$ mv data1 data2.gz
bandit12@bandit:/tmp/pavan$ gzip -d data2.gz
```

Now it’s time to check the retrieved file, we use the file command again. This tells us that it is a bzip2 compressed file.

Now to decompress first, we need to rename the file and provide it with a proper bzip2 extension. We are going to use the move command for this. We renamed the file as data3.bz2. Now using the bzip2 command and -d parameter, we decompress the file.

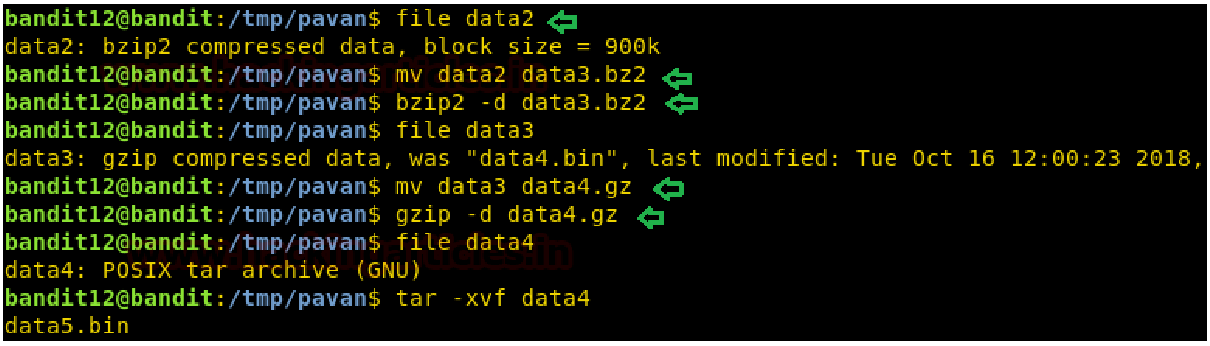
Now it’s time to check the retrieved file, we use the file command again. This tells us that it is a gzip compressed file.

Now decompress first, we need to rename the file and provide it with a proper gzip extension. We are going to use the move command for this. We renamed the file as data4.gz. Now using the gzip command and -d parameter, we decompress the file.

Now it’s time to check the retrieved file, we use the file command again. This tells us that it is a tar archive file.

Now to extract we will use the tar command with xvf parameters. This gives us a file named data5.bin

```
file data2
mv data2 data3.bz2
bzip2 -d data3.bz2
file data3
mv data3 data4.gz
gzip -d data4.gz
file data4
tar -xvf data4
```



```
bandit12@bandit:/tmp/pavan$ file data2
data2: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/pavan$ mv data2 data3.bz2
bandit12@bandit:/tmp/pavan$ bzip2 -d data3.bz2
bandit12@bandit:/tmp/pavan$ file data3
data3: gzip compressed data, was "data4.bin", last modified: Tue Oct 16 12:00:23 2018,
bandit12@bandit:/tmp/pavan$ mv data3 data4.gz
bandit12@bandit:/tmp/pavan$ gzip -d data4.gz
bandit12@bandit:/tmp/pavan$ file data4
data4: POSIX tar archive (GNU)
bandit12@bandit:/tmp/pavan$ tar -xvf data4
data5.bin
```

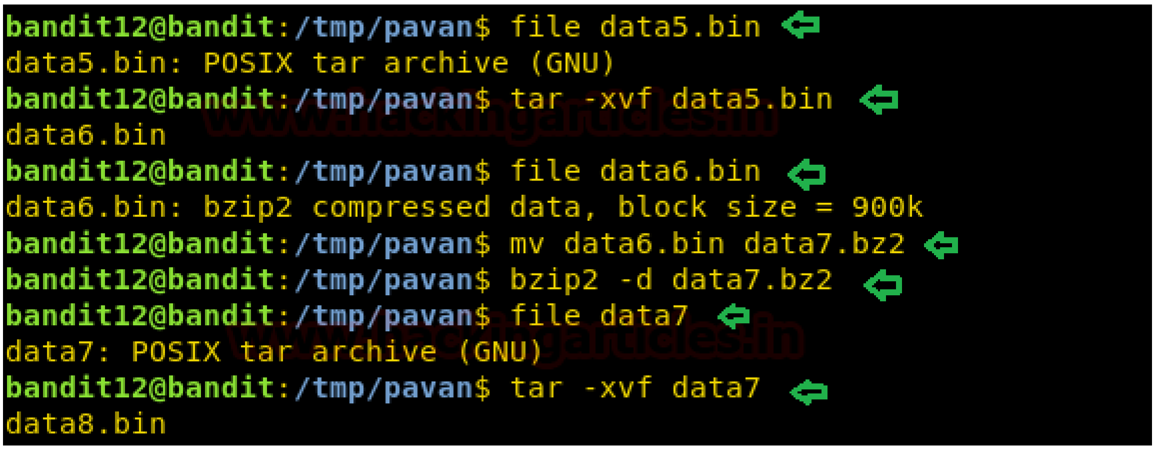
Now it's time to check the retrieved file, we use the file command again. This tells us that it is a tar archive file. Now to extract we will use the tar command with xvf parameters. This gives us a file named data6.bin

Now it's time to check the retrieved file, we use the file command again. This tells us that it is a bzip2 compressed file.

Now decompress first, we need to rename the file and provide it with a proper bzip2 extension. We are going to use the move command for this. We renamed the file as data7.bz2. Now using the bzip2 command and -d parameter, we decompress the file.

Now it's time to check the retrieved file, we use the file command again. This tells us that it is a tar archive file. Now to extract we will use the tar command with xvf parameters. This gives us a file named data8.bin

```
file data5.bin

tar -xvf data5.bin

file data6.bin

mv data6.bin data7.bz2

bzip2 -d data7.bz2

file data7

tar -xvf data7
```



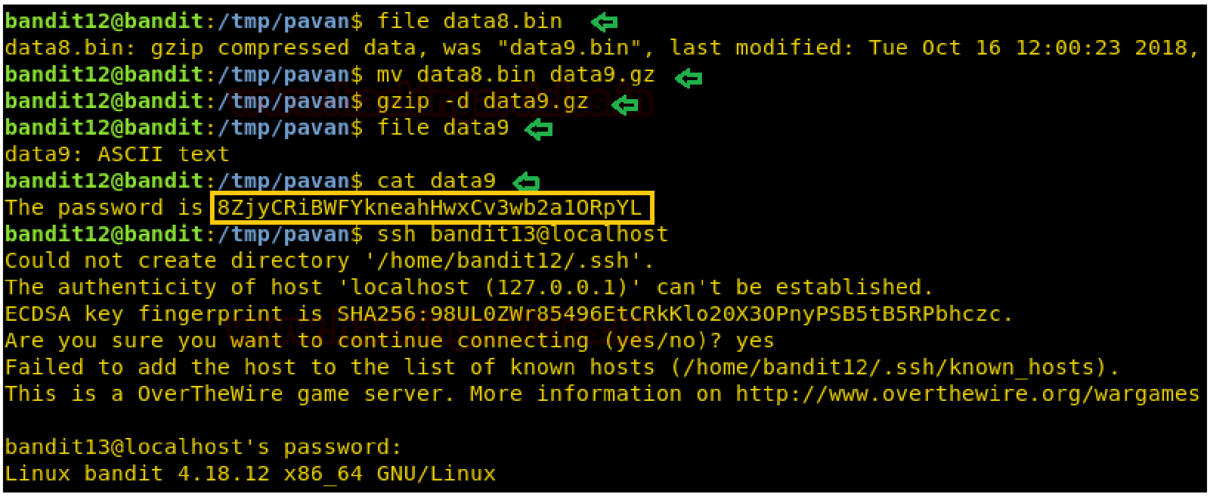
```
bandit12@bandit:/tmp/pavan$ file data5.bin
data5.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/pavan$ tar -xvf data5.bin
data6.bin
bandit12@bandit:/tmp/pavan$ file data6.bin
data6.bin: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/pavan$ mv data6.bin data7.bz2
bandit12@bandit:/tmp/pavan$ bzip2 -d data7.bz2
bandit12@bandit:/tmp/pavan$ file data7
data7: POSIX tar archive (GNU)
bandit12@bandit:/tmp/pavan$ tar -xvf data7
data8.bin
```

Now it's time to check the retrieved file, we use the file command again. This tells us that it is a gzip compressed file.

Now decompress first, we need to rename the file and provide it with a proper gzip extension. We are going to use the move command for this. We renamed the file as data9.gz. Now using the gzip command and -d parameter, we decompress the file.

Now to understand the type of file we are going to use the file command it returns us the type of file. On running the command, we are informed that the file is ASCII text. This might be a readable file. We use the cat command to read the file. This gives us the password for the next level. We will use it to get an SSH connection as bandit13.

```
file data8.bin
mv data8.bin data9.gz
gzip -d data9.gz
file data9
cat data9
ssh bandit13@localhost
```

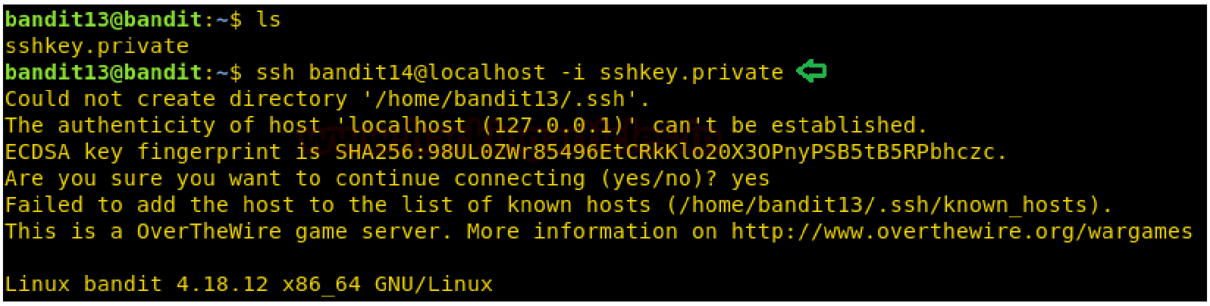


```
bandit12@bandit:/tmp/pavan$ file data8.bin
data8.bin: gzip compressed data, was "data9.bin", last modified: Tue Oct 16 12:00:23 2018,
bandit12@bandit:/tmp/pavan$ mv data8.bin data9.gz
bandit12@bandit:/tmp/pavan$ gzip -d data9.gz
bandit12@bandit:/tmp/pavan$ file data9
data9: ASCII text
bandit12@bandit:/tmp/pavan$ cat data9
The password is 8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL
bandit12@bandit:/tmp/pavan$ ssh bandit13@localhost
Could not create directory '/home/bandit12/.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/bandit12/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit13@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 13-14

We are informed that we are not going to get a password for the next level. Instead, we are given an ssh private key. So, to get to the next level we are going to use that ssh private key. Firstly, let’s find that private key using the ls command. We found the private key. Now we will use it to get an SSH connection as bandit14.

```
ls

ssh bandit14@localhost -i sshkey.private
```

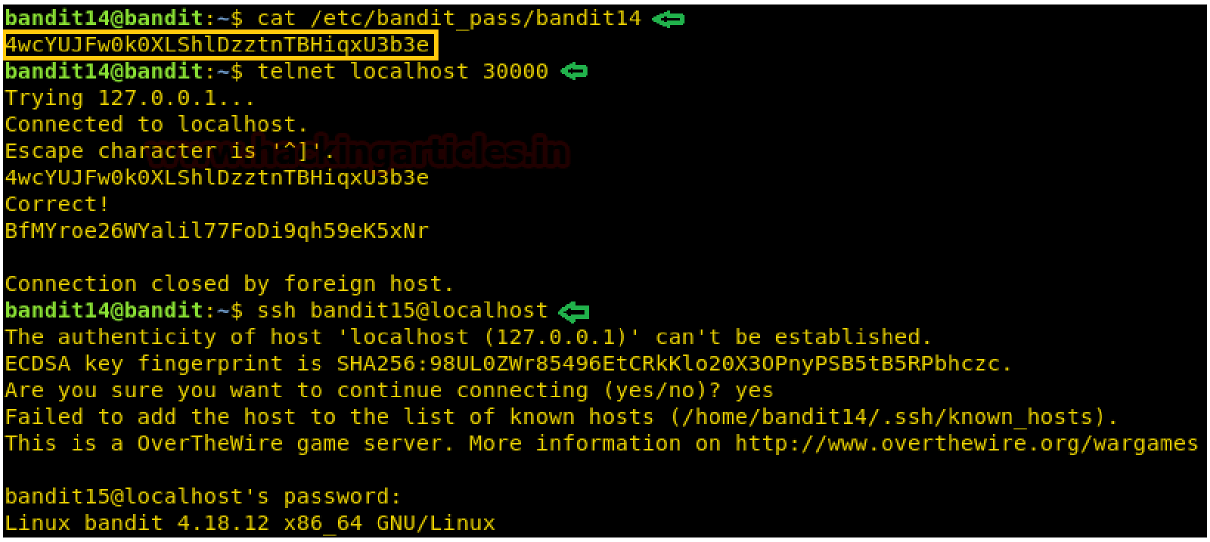


```
bandit13@bandit:~$ ls
sshkey.private
bandit13@bandit:~$ ssh bandit14@localhost -i sshkey.private
Could not create directory '/home/bandit13/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit13/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 14-15

In the previous levels, we got the password for level 14 and have successfully connected as user bandit14. We are informed that the password for the next level can be retrieved by submitting the password of the current level to port 30000 on localhost. First, we retrieve the password for the current level. We used the cat command to print the password as shown in the given image. To connect to port 30000, we are using telnet. After connecting we enter the current password it is checked and upon matching the password for the next level is printed on the screen. We will use this password to get an SSH connection as bandit15

```
cat /etc/bandit_pass/bandit14
telnet localhost 30000
ssh bandit15@localhost
```

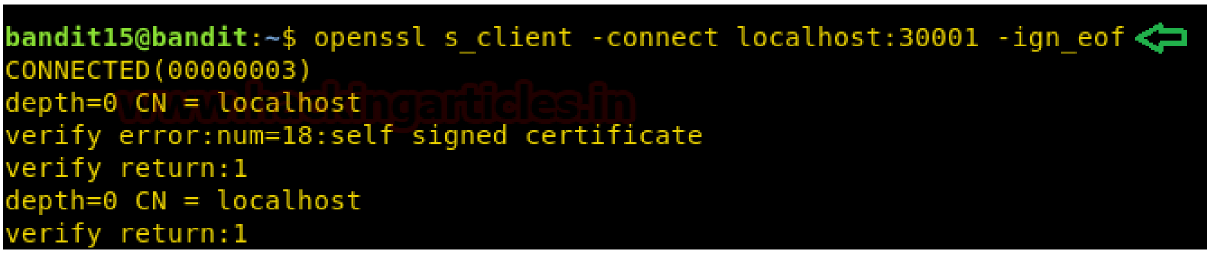


```
bandit14@bandit:~$ cat /etc/bandit_pass/bandit14
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
bandit14@bandit:~$ telnet localhost 30000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
Correct!
BfMYroe26WYalil77FoDi9qh59eK5xNr

Connection closed by foreign host.
bandit14@bandit:~$ ssh bandit15@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)? yes
Failed to add the host to the list of known hosts (/home/bandit14/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit15@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 15-16

On this level, we are informed that the password for the next level is retrieved by submitting the password of the current level to port 30001 on localhost using SSL encryption. We use the openssl command with parameters like s_client that implements that we are the connecting as the client using the hostname localhost at port 30001. We use -ign_eof to inhibit shutting the connection when the end of file is reached in the input.

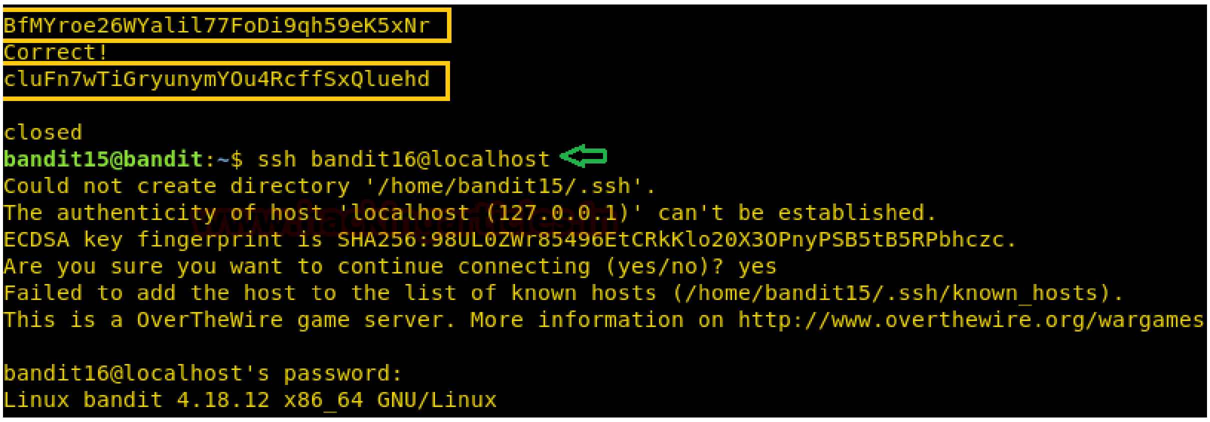
```
openssl s_client -connect localhost:30001 -ign_eof
```



```
bandit15@bandit:~$ openssl s_client -connect localhost:30001 -ign_eof
CONNECTED(00000003)
depth=0 CN = localhost
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = localhost
verify return:1
```

After establishing the connection, we provide it with the password for the bandit15. It is verified and after verification, the password for the next level is provided. We will use this password to get an SSH connection as bandit16.

```
ssh bandit16@localhost
```

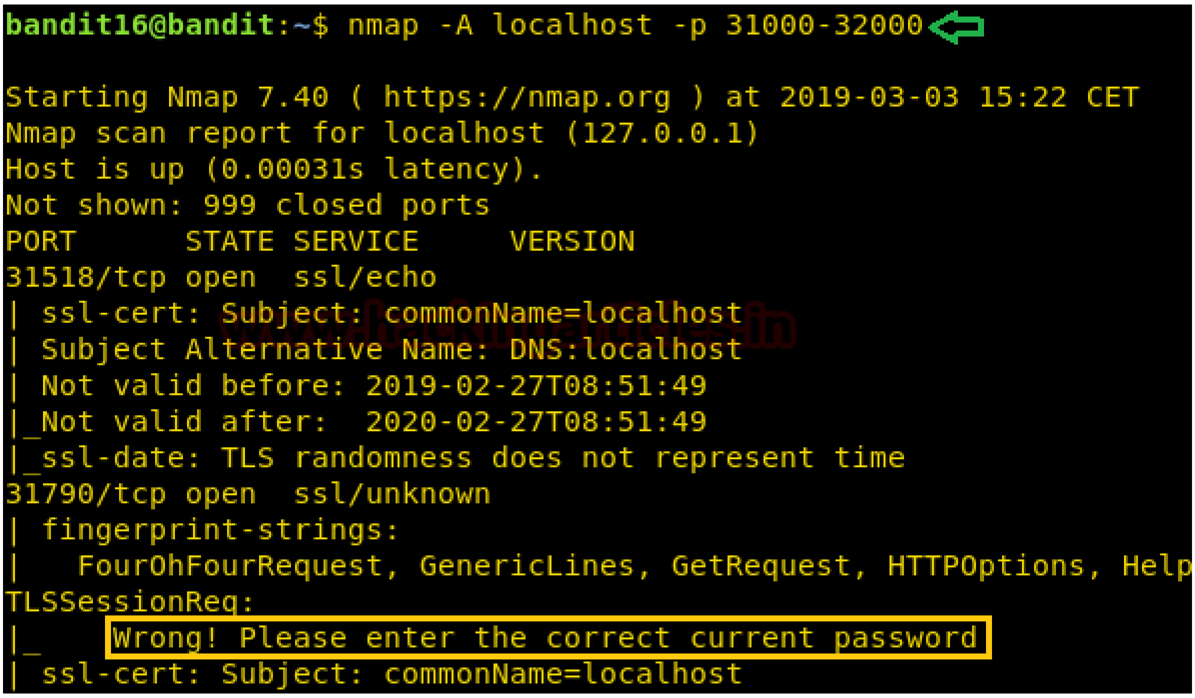


```
BfMYroe26WYalil77FoDi9qh59eK5xNr
Correct!
cluFn7wTiGryunymYOu4RcffSxQluehd

closed
bandit15@bandit:~$ ssh bandit16@localhost
Could not create directory '/home/bandit15/.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/bandit15/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit16@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 16-17

Initially, we are informed that the credentials for the next level can be retrieved by connecting to a port within the range of 31000 to 32000 and submitting the password of bandit16. We use Nmap to scan the ports to get the exact port from the range. As we can see in the output of the Nmap scan that on port 31790 there is a message that hints that we need to enter the password on that port.

```
nmap -A localhost -p 31000-32000
```

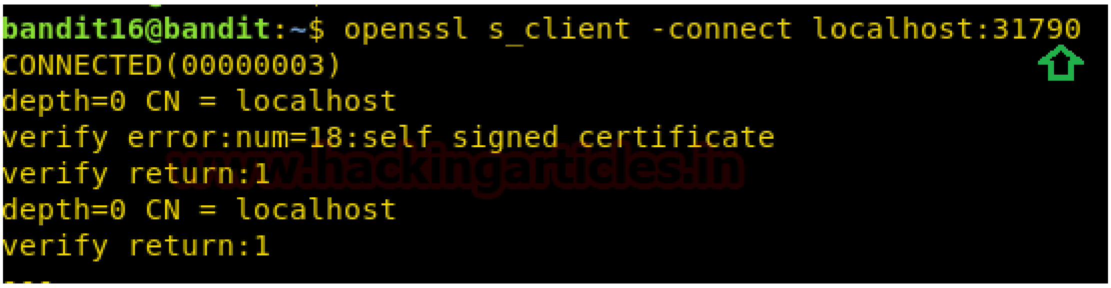


```
bandit16@bandit:~$ nmap -A localhost -p 31000-32000

Starting Nmap 7.40 ( https://nmap.org ) at 2019-03-03 15:22 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00031s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE     VERSION
31518/tcp open  ssl/echo
| ssl-cert: Subject: commonName=localhost
| Subject Alternative Name: DNS:localhost
| Not valid before: 2019-02-27T08:51:49
|_Not valid after:  2020-02-27T08:51:49
|_ssl-date: TLS randomness does not represent time
31790/tcp open  ssl/unknown
| fingerprint-strings:
|   FourOhFourRequest, GenericLines, GetRequest, HTTPOptions, Help
TLSSessionReq:
|_    Wrong! Please enter the correct current password
| ssl-cert: Subject: commonName=localhost
```

Now we will connect to this port using openssl as localhost.

```
openssl s_client -connect localhost:31790
```

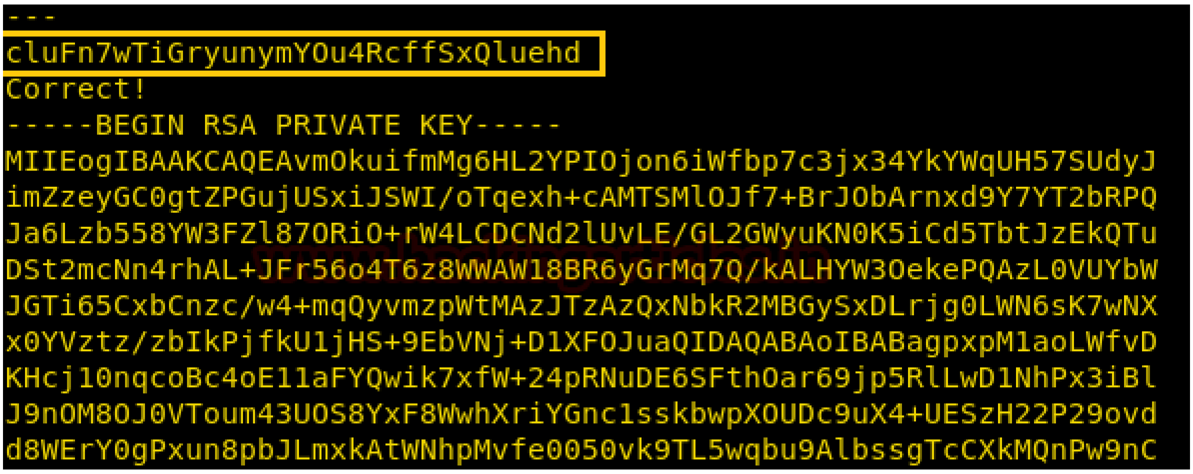
After connecting to the port, we will have to enter the password of bandit16. This password goes under verification. Upon a successful match, we are provided with an RSA key.



```
bandit16@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
```

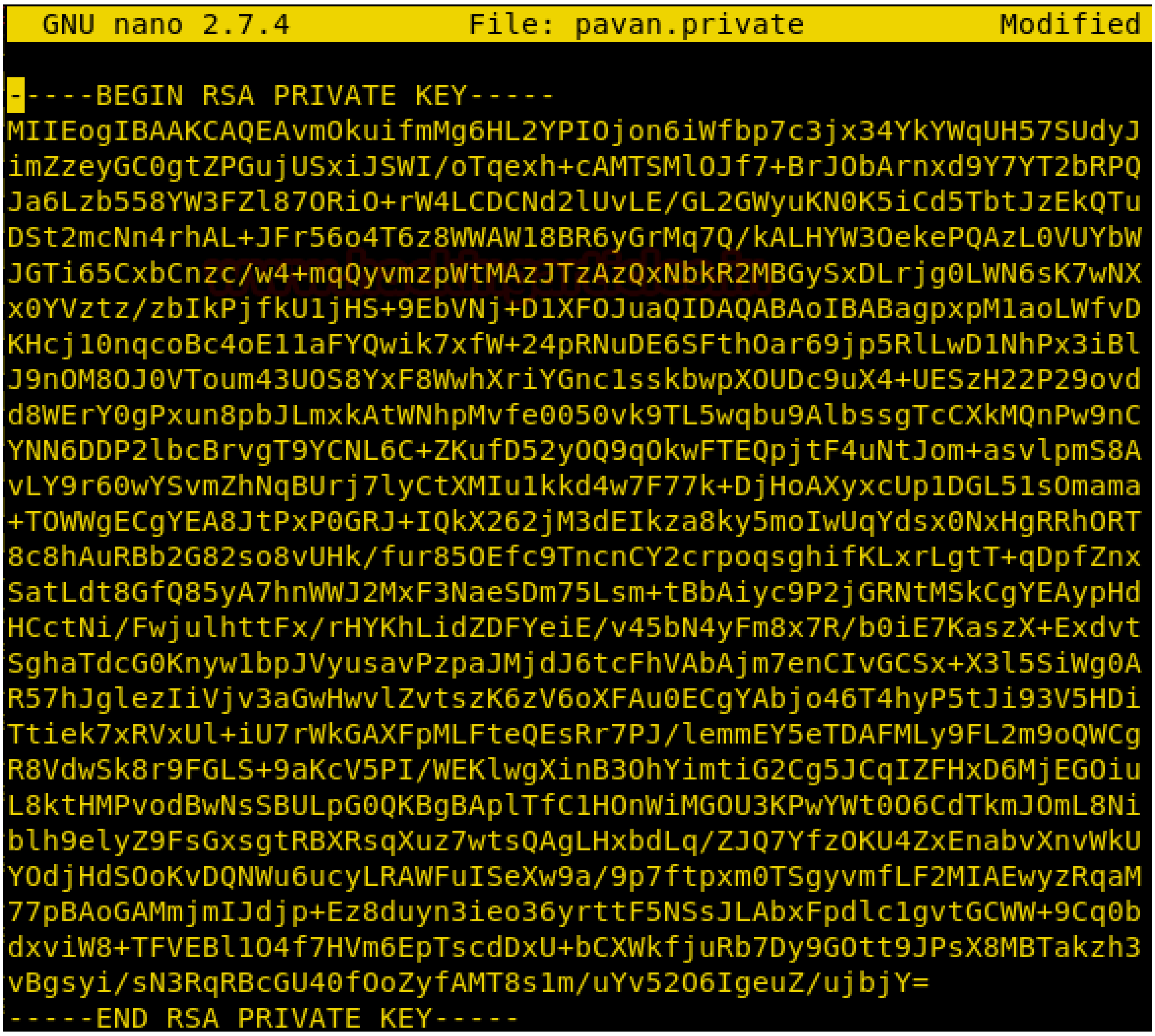
Now to use this RSA key, we need to create a private key. But we can't do this inside the home directory as we lack necessary permissions. So, we create a directory in /tmp directory using mkdir command. On traversing to that newly created directory, we will create a private key. We can name it anything we want. Here we are using the nano editor to create the private key.

```
mkdir /tmp/pavan_ssh

cd /tmp/pavan_ssh

nano pavan.private
```



```
---
cluFn7wTiGryunymYOu4RcffSxQluehd
Correct!
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAvmOkuifmMg6HL2YPIOjon6iWfbp7c3jx34YkYWqUH57SUdyJ
imZzeyGC0gtZPGujUSxiJSWI/oTqexh+cAMTSMlOJf7+BrJObArnxd9Y7YT2bRPQ
Ja6Lzb558YW3FZl87ORiO+rW4LCDCNd2lUvLE/GL2GWyuKN0K5iCd5TbtJzEkQTu
DSt2mcNn4rhAL+JFr56o4T6z8WWAW18BR6yGrMq7Q/kALHYW3OekePQAzL0VUYbW
JGTi65CxbCnzc/w4+mqQyvmzpWtMAzJTzAzQxNbkR2MBGySxDLrjg0LWN6sK7wNX
x0YVztz/zbIkPjfkU1jHS+9EbVNj+D1XFOJuaQIDAQABAoIBABagpxpM1aoLWfvD
KHcj10nqcoBc4oE1laFYQwik7xfW+24pRNuDE6SFthOar69jp5RlLwD1NhPx3iBl
J9nOM8OJ0VToum43UOS8YxF8WwhXriYGnclsskbwpXOUDc9uX4+UESzH22P29ovd
d8WErY0gPxun8pbJLmxkAtWNhpMvfe0050vk9TL5wqbu9AlbssgTcCXkMQnPw9nC
```

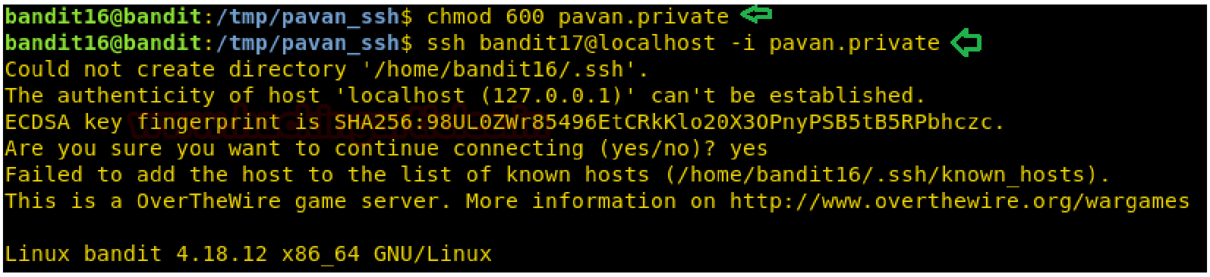
After running the nano command, we will be prompted to press the Enter key to continue. On doing that the private key will be opened to edit using nano. Now we will paste the RSA key we found earlier. Now to exit we will press Ctrl and x keys simultaneously. There would be a prompt asking us to save the updates. We will press 'y' followed by this, nano will ask us if we want to rename the file. After this, we would have successfully created a private key using the RSA we were provided before.



```
  GNU nano 2.7.4              File: pavan.private              Modified

-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAvmOkuifmMg6HL2YPIOjon6iWfbp7c3jx34YkYWqUH57SUdyJ
imZzeyGC0gtZPGujUSxiJSWI/oTqexh+cAMTSMlOJf7+BrJObArnxd9Y7YT2bRPQ
Ja6Lzb558YW3FZl87ORiO+rW4LCDCNd2lUvLE/GL2GWyuKN0K5iCd5TbtJzEkQTu
DSt2mcNn4rhAL+JFr56o4T6z8WWAW18BR6yGrMq7Q/kALHYW3OekePQAzL0VUYbW
JGTi65CxbCnzc/w4+mqQyvmzpWtMAzJTzAzQxNbkR2MBGySxDLrjg0LWN6sK7wNX
x0YVztz/zbIkPjfkU1jHS+9EbVNj+D1XFOJuaQIDAQABAoIBABagpxpM1aoLWfvD
KHcj10nqcoBc4oE1laFYQwik7xfW+24pRNuDE6SFthOar69jp5RlLwD1NhPx3iBl
J9nOM8OJ0VToum43UOS8YxF8WwhXriYGnclsskbwpXOUDc9uX4+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
R8VdwSk8r9FGLS+9aKcV5PI/WEKlwgXinB3OhYimtiG2Cg5JCqIZFHxD6MjEGOiu
L8ktHMPvodBwNsSBULpG0QKBgBAplTfC1HOnWiMGOU3KPwYWt0O6CdTkmJOmL8Ni
blh9elyZ9FsGxsgtRBXRsqXuz7wtsQAgLHxbdLq/ZJQ7YfzOKU4ZxEnabvXnvWkU
YOdjHdSOoKvDQNWu6ucyLRAWFuISeXw9a/9p7ftpxm0TSgyvmfLF2MIAEwyzRqaM
77pBAoGAMmjmIJdjp+Ez8duyn3ieo36yrttF5NSsJLAbxFpdlc1gvtGCWW+9Cq0b
dxviW8+TFVEBl1O4f7HVm6EpTscdDxU+bCXWkfjuRb7Dy9GOtt9JPsX8MBTakzh3
vBgsyi/sN3RqRBcGU40fOoZyfAMT8s1m/uYv52O6IgeuZ/ujbjY=
-----END RSA PRIVATE KEY-----
```

SSH won’t allow any private key with such open permissions. So, we will have to change the permissions. We will use the chmod command to apply the permissions equivalent to 600. This means that only the owner can read and write the file. We will use this private key to get an SSH connection as bandit17.

```
chmod 600 pavan.private
ssh bandit17@localhost -i pavan.private
```

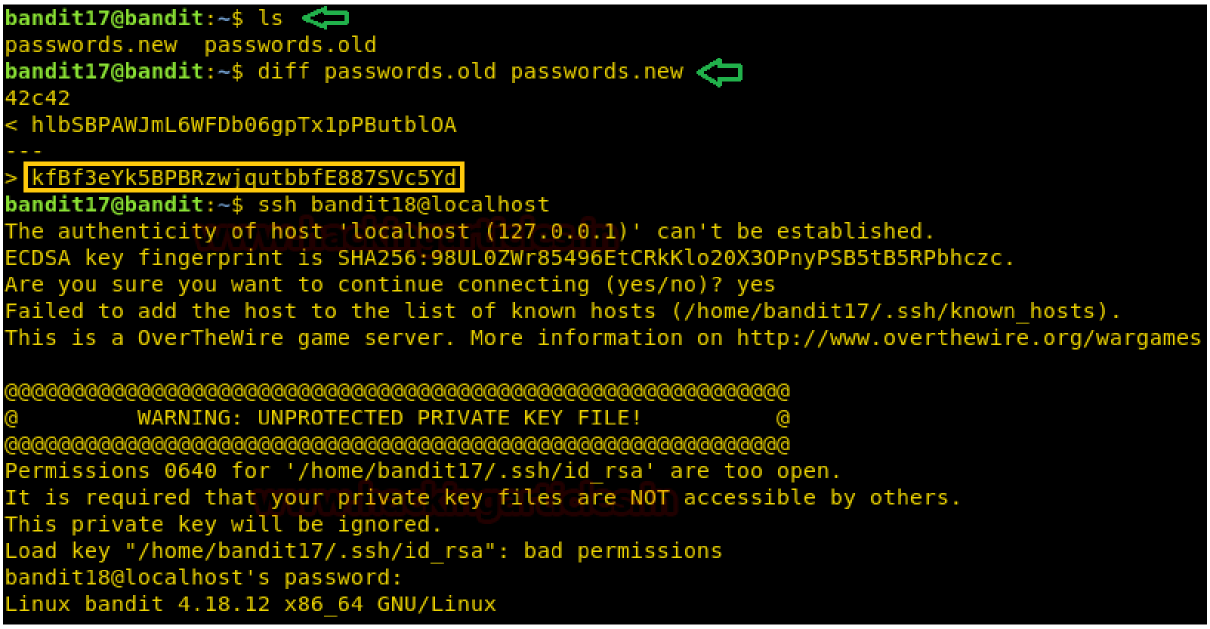


```
bandit16@bandit:/tmp/pavan_ssh$ chmod 600 pavan.private
bandit16@bandit:/tmp/pavan_ssh$ ssh bandit17@localhost -i pavan.private
Could not create directory '/home/bandit16/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit16/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 17-18

Upon logging in as bandit17, we run the ls command to look for any files. We see that we have two files, password.new and password.old. Now we have informed that password for the next level the only line that has been changed between both files. We will use the diff command to find that password. And the diff command gives us the required password. We will use this password to get an SSH connection as bandit18.

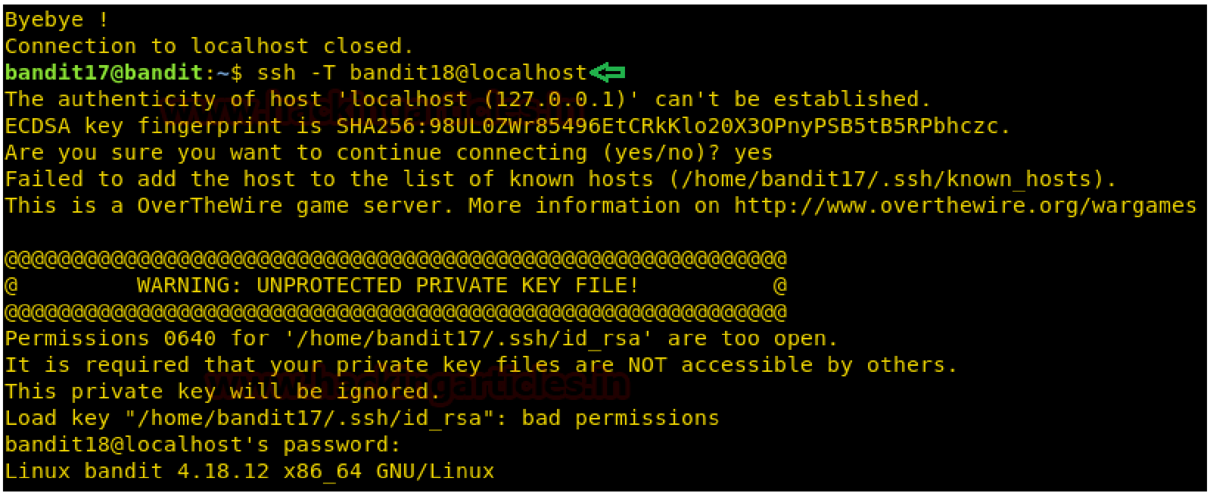
```
ls
diff passwords.old passwords.new
ssh bandit18@localhost
```



```
bandit17@bandit:~$ ls
passwords.new  passwords.old
bandit17@bandit:~$ diff passwords.old passwords.new
42c42
< hlbSBPAWJmL6WFDb06gpTx1pPButblOA
---
> kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd
bandit17@bandit:~$ ssh bandit18@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)? yes
Failed to add the host to the list of known hosts (/home/bandit17/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0640 for '/home/bandit17/.ssh/id_rsa' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "/home/bandit17/.ssh/id_rsa": bad permissions
bandit18@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Now on providing with the correct password our connection was closed. This is because the authors of this level have modified the .bashrc file to log us out of ssh. We will use the -t parameter to disable the pseudo -tty allocation. As this is making our session vulnerable to get closed. Let's connect ssh again as shown in the given image.

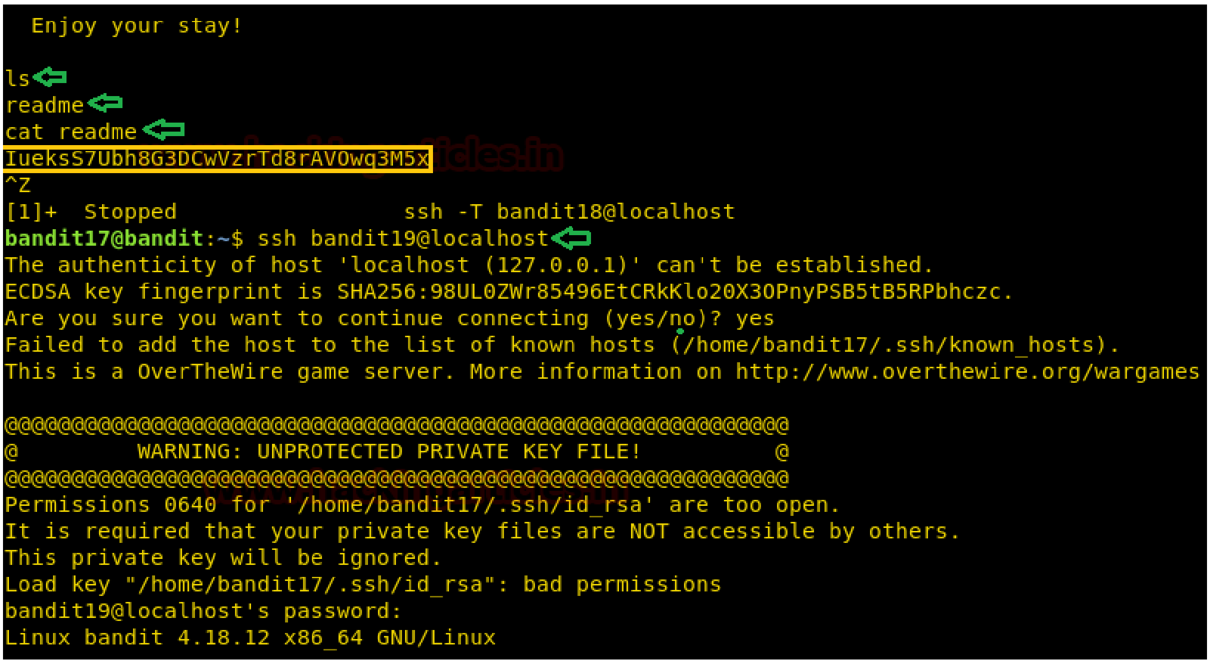
```
ssh -T bandit18@localhost
```



```
Byebye !
Connection to localhost closed.
bandit17@bandit:~$ ssh -T bandit18@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)? yes
Failed to add the host to the list of known hosts (/home/bandit17/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0640 for '/home/bandit17/.ssh/id_rsa' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "/home/bandit17/.ssh/id_rsa": bad permissions
bandit18@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

This time we got a shell, it may be not visible but it is there. We can run commands here. First, let's try the ls command. This gives us the readme file. Upon reading that file, we get what seems like credentials for the next level. We will use this password to get an SSH connection as bandit19.

```
ls
cat readme
ssh bandit19@localhost
```

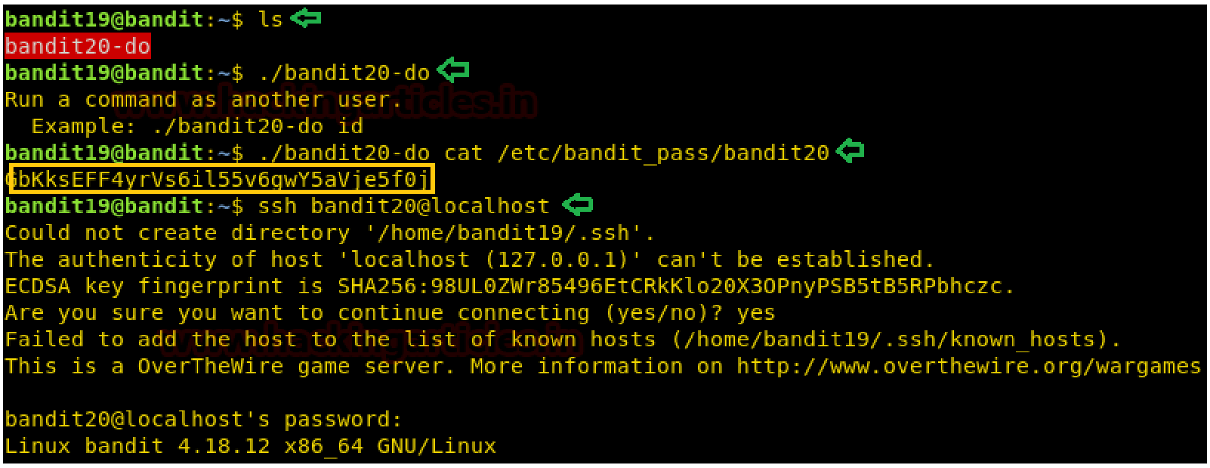


```
  Enjoy your stay!

ls
readme
cat readme
IueksS7Ubh8G3DCwVzrTd8rAVOwq3M5x
^Z
[1]+  Stopped                 ssh -T bandit18@localhost
bandit17@bandit:~$ ssh bandit19@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)? yes
Failed to add the host to the list of known hosts (/home/bandit17/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0640 for '/home/bandit17/.ssh/id_rsa' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "/home/bandit17/.ssh/id_rsa": bad permissions
bandit19@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 19-20

After successfully getting the ssh to user bandit19, we start with ls command to see what we got this time. We have a file that seems like a script. We tried to run to see the working of the script. We are shown that the script runs a command as another user. Now we were informed that the password is stored at /etc/bandit_pass/. So, we run the script with the cat command to read the password for the next level. We will use this password to get an SSH connection as bandit20.

```
ls
./bandit20-do
./bandit20-do cat /etc/bandit_pass/bandit20
ssh bandit20@localhost
```



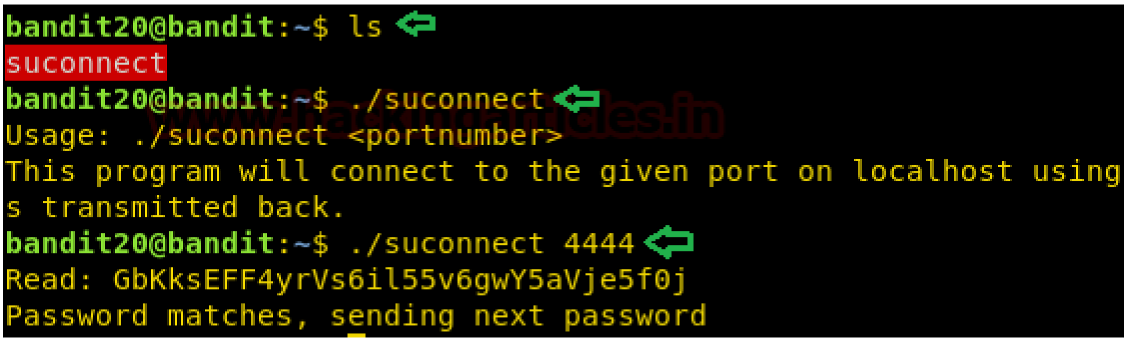
```
bandit19@bandit:~$ ls
bandit20-do
bandit19@bandit:~$ ./bandit20-do
Run a command as another user.
  Example: ./bandit20-do id
bandit19@bandit:~$ ./bandit20-do cat /etc/bandit_pass/bandit20
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
bandit19@bandit:~$ ssh bandit20@localhost
Could not create directory '/home/bandit19/.ssh'.
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/bandit19/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit20@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 20-21

We are informed that there is a setuid binary in this level whose job is to make a connection to localhost on a port and read the password used to login as bandit20 and then send the password for the next level. First, let's see the files we have using the command ls. We have a script suconnect. On running this command without any parameters, we see that it requires a port to connect to. Now here is the part where it gets tricky. The image given below is one instance of the shell. We will execute to the point where we run suconnect without parameters and create other instance of the same shell. Run a netcat listener over another instance on the same port we are planning to suconnect. But we need to start listener before running the suconnect. On running the suconnect. Netcat will grab a session. Now we enter the password that we used to login as user bandit20. As we can see that the password, we entered is read by the suconnect and when the password is verified. Password for the next level is sent to the listener.

```
ls
./suconnect
./suconnect 4444
```

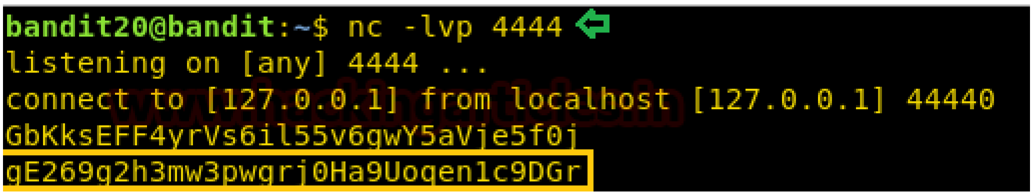
Image shown below is the execution of the first instance.



```
bandit20@bandit:~$ ls
suconnect
bandit20@bandit:~$ ./suconnect
Usage: ./suconnect <portnumber>
This program will connect to the given port on localhost using
s transmitted back.
bandit20@bandit:~$ ./suconnect 4444
Read: GbKksEFF4yrVs6il55v6gwY5aVje5f0j
Password matches, sending next password
```

```
nc -lvp 4444
```

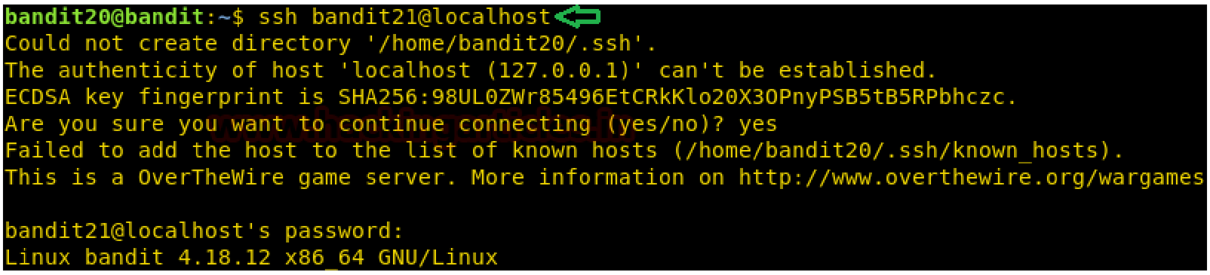
Image shown below is the execution of the second instance.



```
bandit20@bandit:~$ nc -lvp 4444
listening on [any] 4444 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 44440
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr
```

Now that we have the password for the next level, we move back to our first instance and used the password to login as user bandit21 using SSH.

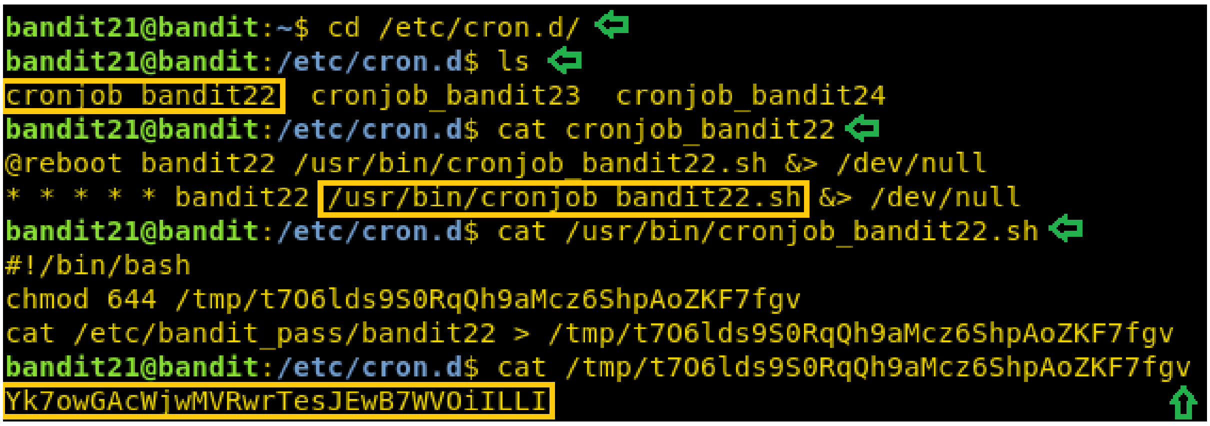
```
ssh bandit21@localhost
```



```
bandit20@bandit:~$ ssh bandit21@localhost
Could not create directory '/home/bandit20/.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/bandit20/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit21@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

In the previous levels, we got the password for level 21 and have successfully connected as user bandit21. We are informed that there is a cron script running and we need to enumerate /etc/cron.d/ for the password. So, we traversed to that path. We use ls command to show the list of files inside the directory. As the next level is bandit22 so we read the cronjob_bandit22 using cat command. It shows that there is a script at /usr/bin/cronjob_bandit22.sh. So, we read that script to find that it writes the password for the next user inside a file that is located inside the tmp directory. On reading that file we got the password we required to get on to the next level.

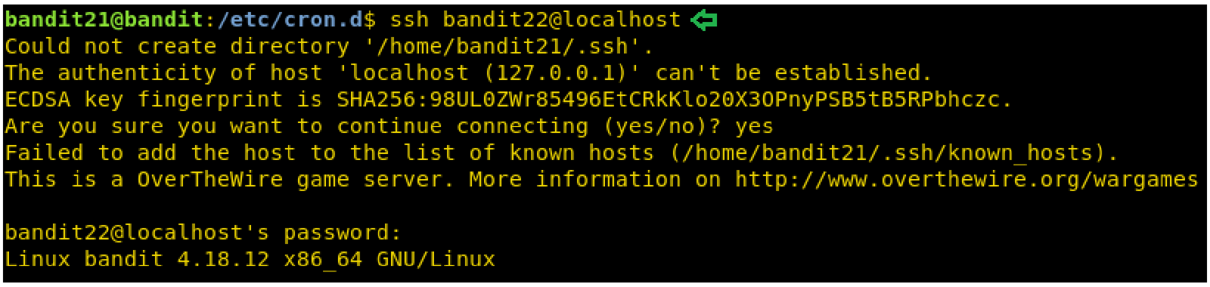
```
cd /etc/cron.d/
ls
cat cronjob_bandit22
cat /usr/bin/cronjob_bandit22.sh
cat /tmp/t706lds9S0RqQh9aMcz6ShpAoZKF7fgv
```



```
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$ cat /usr/bin/cronjob_bandit22.sh
#!/bin/bash
chmod 644 /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
cat /etc/bandit_pass/bandit22 > /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
bandit21@bandit:/etc/cron.d$ cat /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI
```

Now that we have the password for the next level, we will login as bandit22 using SSH.

```
ssh bandit22@localhost
```

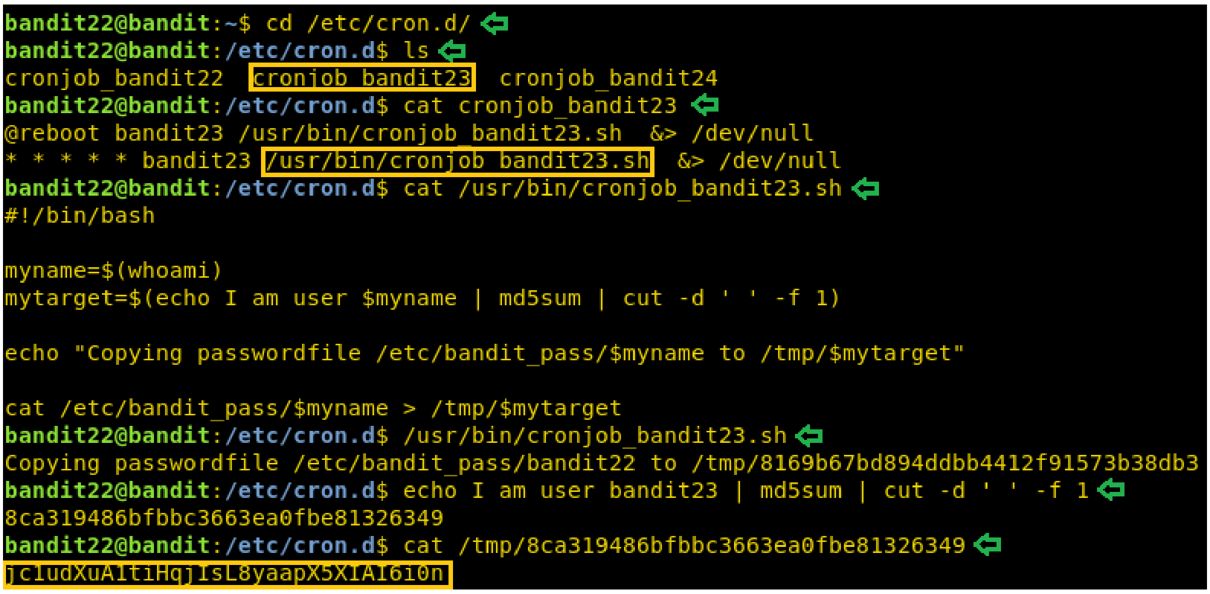


```
bandit21@bandit:/etc/cron.d$ ssh bandit22@localhost
Could not create directory '/home/bandit21/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit21/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit22@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 22-23

On this level, we are informed that there is a cron script running and we need to enumerate /etc/cron.d/ for the password. So, we traversed to that path. We use ls command to show the list of files inside the directory. As the next level is bandit23 so we read the cronjob_bandit23 using cat command. It shows that there is a script at /usr/bin/cronjob_bandit23.sh. So, we read that script using cat command. This script has a variable called myname which is the output of the command whoami. Which basically return bandit22. Next, the operation is done on this variable. It prints “I am user bandit22” and it is encrypted in MD5. This hash is used to name the file which has the password for the next level. Now to get the password for the bandit23 user, we run the command with the value for the variable myname set to bandit23. This will give us the hash value which further gives the name of the file in the tmp directory.

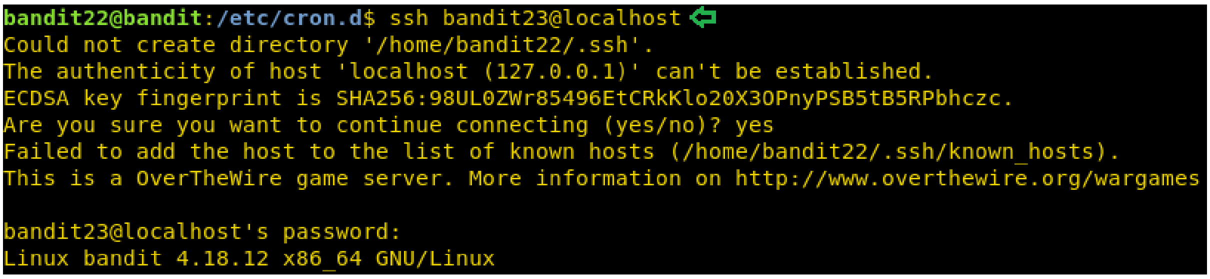
```
cd /etc/cron.d/
ls
cat cronjob_bandit23
cat /usr/bin/cronjob_bandit23.sh
/usr/bin/cronjob_bandit23.sh
echo I am user bandit23 | md5sum | cut -d ' ' -f 1
cat /tmp/8ca319486bfbbc3663ea0fbe81326349
```



```
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$ /usr/bin/cronjob_bandit23.sh
Copying passwordfile /etc/bandit_pass/bandit22 to /tmp/8169b67bd894ddbb4412f91573b38db3
bandit22@bandit:/etc/cron.d$ echo I am user bandit23 | md5sum | cut -d ' ' -f 1
8ca319486bfbbc3663ea0fbe81326349
bandit22@bandit:/etc/cron.d$ cat /tmp/8ca319486bfbbc3663ea0fbe81326349
jcludXuA1tiHqjIsL8yaapX5XIAI6i0n
```

Now that we have the password for the next level, we will login as bandit23 using SSH.

```
ssh bandit23@localhost
```

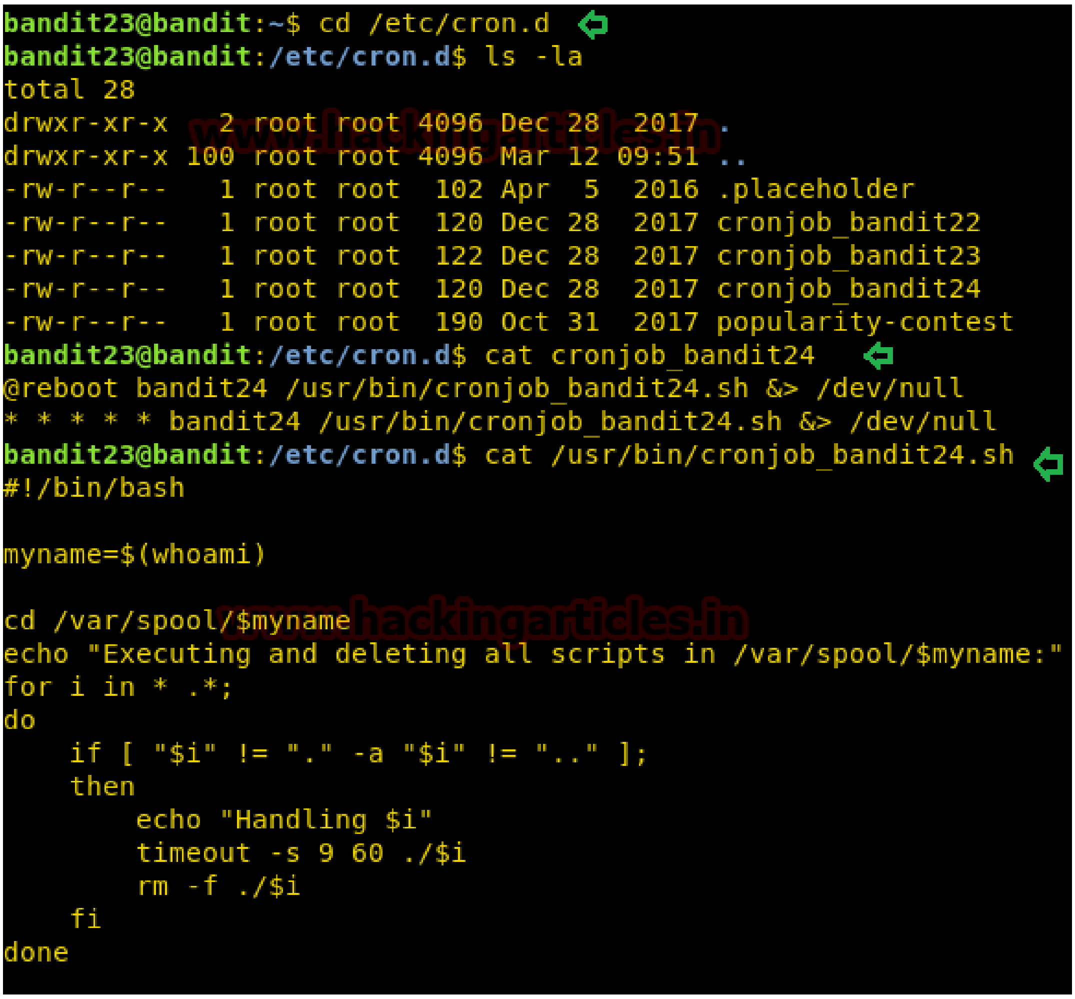


```
bandit22@bandit:/etc/cron.d$ ssh bandit23@localhost
Could not create directory '/home/bandit22/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit22/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit23@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 23-24

On this level, we are informed that there is a cron script running and we need to enumerate /etc/cron.d/ for the password. So, we traversed to that path. We use ls command to show the list of files inside the directory. As the next level is bandit24 so we read the cronjob_bandit24 using cat command. It shows that there is a script at /usr/bin/cronjob_bandit24.sh. So, we read that script using cat command. We see that we have a script with a variable named myname which consists of the output of the whoami command. The script first changes the name directory to /var/spool and then executes files with the variable myname file. And after executing it deletes all files inside that directory.

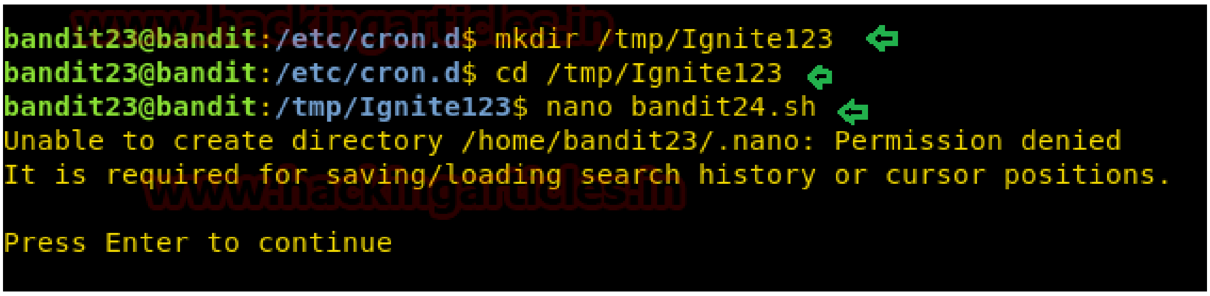
```
cd /etc/cron.d/
ls -la
cat cronjob_bandit24
cat /usr/bin/cronjob_bandit24.sh
```



```
bandit23@bandit:~$ cd /etc/cron.d
bandit23@bandit:/etc/cron.d$ ls -la
total 28
drwxr-xr-x   2 root root 4096 Dec 28  2017 .
drwxr-xr-x 100 root root 4096 Mar 12 09:51 ..
-rw-r--r--   1 root root  102 Apr  5  2016 .placeholder
-rw-r--r--   1 root root  120 Dec 28  2017 cronjob_bandit22
-rw-r--r--   1 root root  122 Dec 28  2017 cronjob_bandit23
-rw-r--r--   1 root root  120 Dec 28  2017 cronjob_bandit24
-rw-r--r--   1 root root  190 Oct 31  2017 popularity-contest
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
```

Now to get the password for the next directory we will have to create a script of our own so that we can put it inside the /var/spool that will cat the password file from the /etc/bandit_pass/bandit24. We will have to save the file with the name of the next user in order to run the file as a cron job successfully.

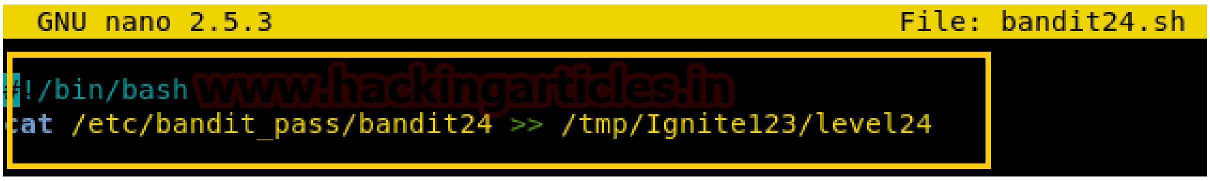
```
mkdir /tmp/Ignite123
cd /tmp/Ignite123
nano bandit24.sh
```



```
bandit23@bandit:/etc/cron.d$ mkdir /tmp/Ignite123
bandit23@bandit:/etc/cron.d$ cd /tmp/Ignite123
bandit23@bandit:/tmp/Ignite123$ 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
```

After creating a file using nano, we will write the script that will read the password from the /etc/bandit_pass and writes in the file inside the directory we just created.

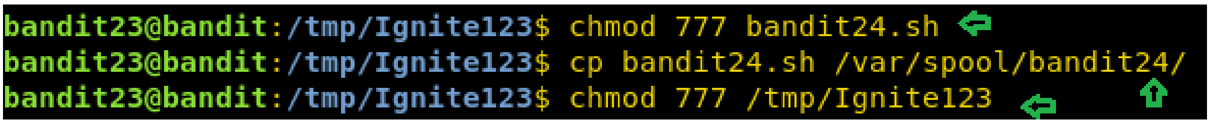
```
#!/bin/bash
cat /etc/bandit_pass/bandit24 >> /tmp/Ignite123/level24
```



```
  GNU nano 2.5.3                                    File: bandit24.sh

#!/bin/bash
cat /etc/bandit_pass/bandit24 >> /tmp/Ignite123/level24
```

Now to execute successfully, we will have to give proper read and write permissions to the script we just created and also to the directory we created.

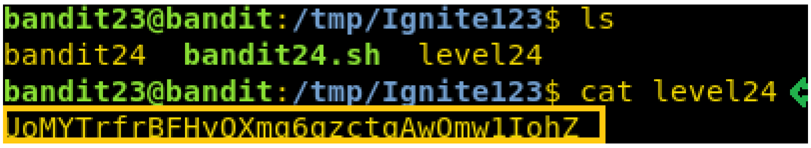
```
chmod 777 bandit24.sh
cp bandit24.sh /var/spool/bandit24/
chmod 777 /tmp/Ignite123
```



```
bandit23@bandit:/tmp/Ignite123$ chmod 777 bandit24.sh
bandit23@bandit:/tmp/Ignite123$ cp bandit24.sh /var/spool/bandit24/
bandit23@bandit:/tmp/Ignite123$ chmod 777 /tmp/Ignite123
```

We will have to wait for some time. We got a bit stuck here as we didn't wait for enough. Have a bit of patience, it will take some time. After that when we list the files inside the directory, we see that a new file is created and upon reading the contents of that file, we find the password that we were looking for in this level. Now that we have the password for the next level, we will login as bandit24 using SSH.

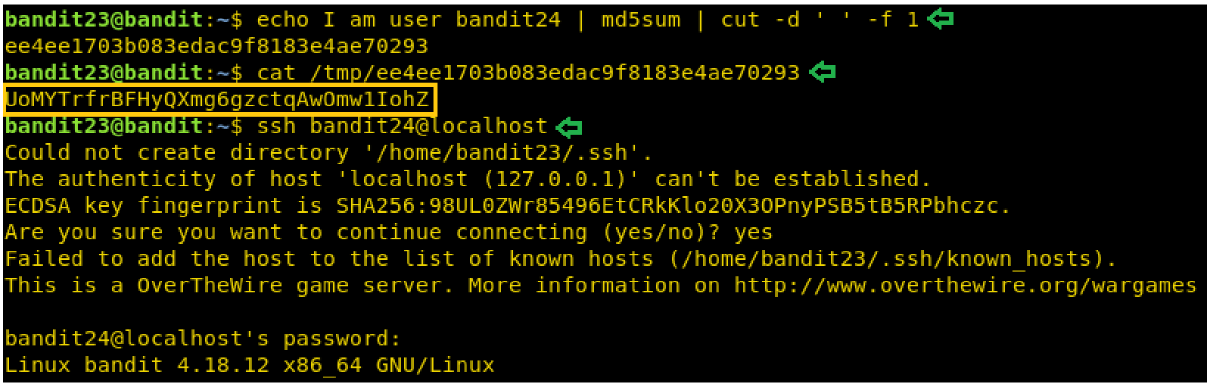
```
ls
cat level24
```



```
bandit23@bandit:/tmp/Ignite123$ ls
bandit24  bandit24.sh  level24
bandit23@bandit:/tmp/Ignite123$ cat level24
UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ
```

Now, if the above-mentioned method doesn't work for you. This is another method to grab the password. It is based on the method that we did at an earlier level. In the previous level we wrote the I am user bandit23 now that we have to grab the password for bandit24 we will write I am user bandit24 and convert it to MD5 and use that text as a directory for the password for the next level. We prefer this method because is obviously faster and easier.

```
echo I am user bandit24 | md5sum | cut -d ' ' -f 1
cat /tmp/ee4ee1703b083edac9f8183e4ae70293
ssh bandit24@localhost
```

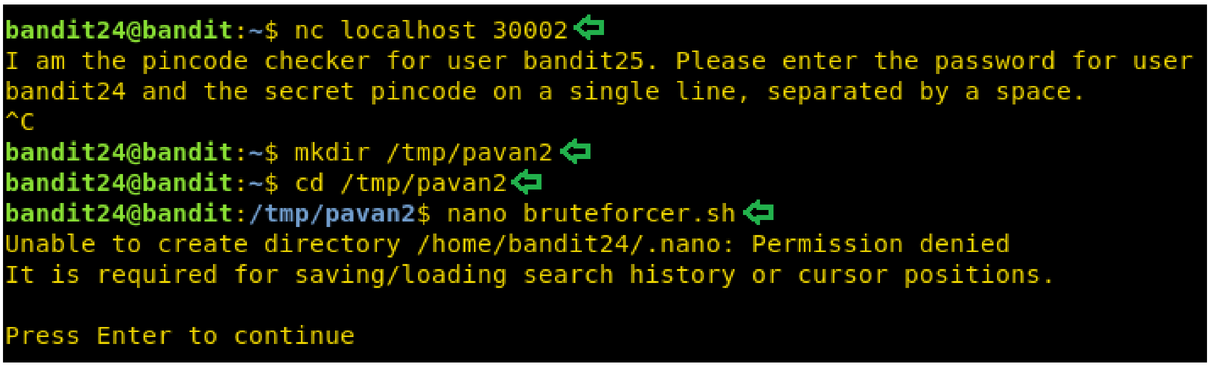


```
bandit23@bandit:~$ echo I am user bandit24 | md5sum | cut -d ' ' -f 1
ee4ee1703b083edac9f8183e4ae70293
bandit23@bandit:~$ cat /tmp/ee4ee1703b083edac9f8183e4ae70293
UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ
bandit23@bandit:~$ ssh bandit24@localhost
Could not create directory '/home/bandit23/.ssh'.
The authenticity 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/bandit23/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit24@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 24-25

On this level, we are informed that a background process is running. It is listening at post 30002 and will give the password for the next level. And we will have to feed it the password for the current level. But wait there is a catch. We will also have to provide a 4-digit secret passcode which will have to Bruteforce as we have absolutely no clue about it. Now to apply Bruteforce we will have to create a Dictionary. As always, we will be needing to read and write permissions to create a script. So, we will create a directory inside the tmp directory. Let’s create a script using nano.

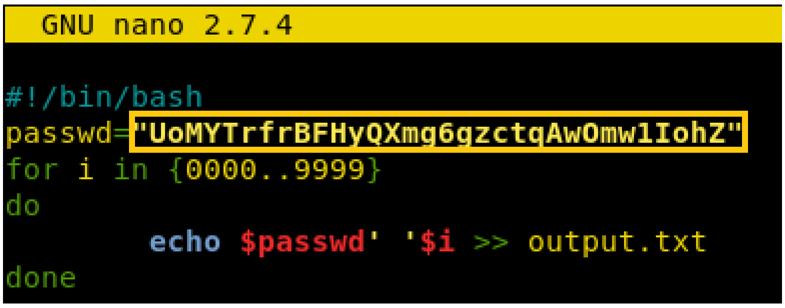
```
nc localhost 30002
cd /tmp/pavan2
nano bruteforcer.sh
```



```
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.
^C
bandit24@bandit:~$ mkdir /tmp/pavan2
bandit24@bandit:~$ cd /tmp/pavan2
bandit24@bandit:/tmp/pavan2$ nano bruteforcer.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
```

After creating the script file, we will have to create a file that would act as a dictionary. We are told that we will have to feed the daemon running on port 30002 the password of the current level followed by a 4-digit passcode. So, we ran a loop that lists all the 4 digits and writes those inside a file called output. This file will act as a dictionary.

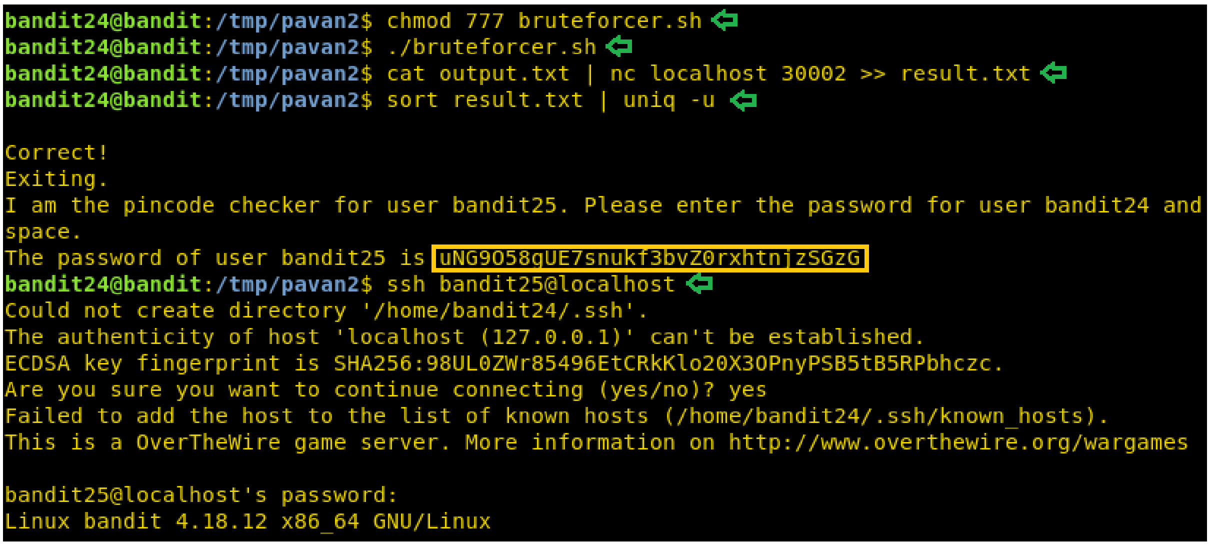
```
#!/bin/bash
passwd="UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ"
for i in {8000..8999}
do
echo $passwd' '$i >> output.txt
done
```



```
  GNU nano 2.7.4

#!/bin/bash
passwd="UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ"
for i in {0000..9999}
do
        echo $passwd' '$i >> output.txt
done
```

Now before running the above script, let’s first give it proper permissions. After that, we will run the script. Now, to apply Bruteforce, we will have to use piping (|). We will first read the password we created inside the output.txt than we will feed its output to the nc at 30002 port. Further, we will feed the output to a file called result. This will make reading the password easier. Now using the sort command combined with the uniq command, we will extract the correct password easily. Now that we have the password for the next level, we will login as bandit25 using SSH.

```
chmod 777 bruteforcer.sh
./bruteforcer.sh
cat output.txt | nc localhost 30002 >> result.txt
sort result.txt | uniq -u
ssh bandit25@localhost
```



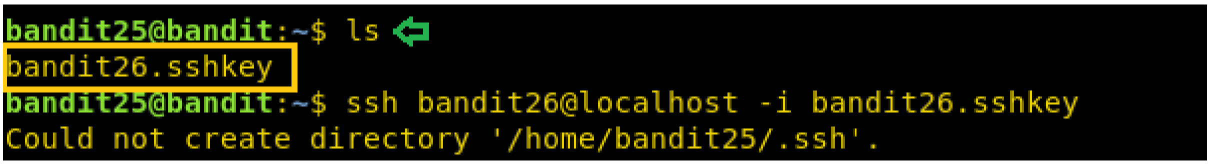
```
bandit24@bandit:/tmp/pavan2$ chmod 777 bruteforcer.sh
bandit24@bandit:/tmp/pavan2$ ./bruteforcer.sh
bandit24@bandit:/tmp/pavan2$ cat output.txt | nc localhost 30002 >> result.txt
bandit24@bandit:/tmp/pavan2$ sort result.txt | uniq -u

Correct!
Exiting.
I am the pincode checker for user bandit25. Please enter the password for user bandit24 and
space.
The password of user bandit25 is uNG9O58gUE7snukf3bvZ0rxhtnjzSGzG
bandit24@bandit:/tmp/pavan2$ ssh bandit25@localhost
Could not create directory '/home/bandit24/.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/bandit24/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit25@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Note: When we were trying the Bruteforce, there were times when we were getting a session timeout error. To resolve this, don't Bruteforce from 0 to 9999. Instead, divide the dictionary into small sections like 0 to 1000 and 1001 to 2000 and so on.

Level 25-26

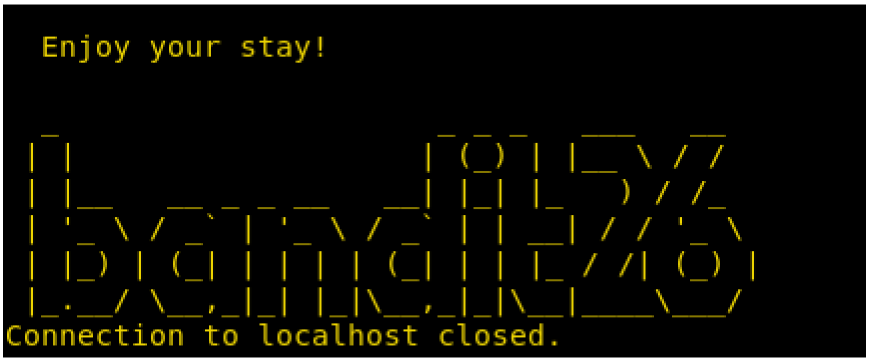
On this level, we are informed that the shell for user bandit26 is not bin bash. So, we will have to figure it out. After logging in as bandit25, we ran the ls command to list all the files inside the directory. This gave an ssh key. So, we tried to login with it.

```
ls
ssh bandit26@localhost -i bandit26.sshkey
```



```
bandit25@bandit:~$ ls
bandit26.sshkey
bandit25@bandit:~$ ssh bandit26@localhost -i bandit26.sshkey
Could not create directory '/home/bandit25/.ssh'.
```

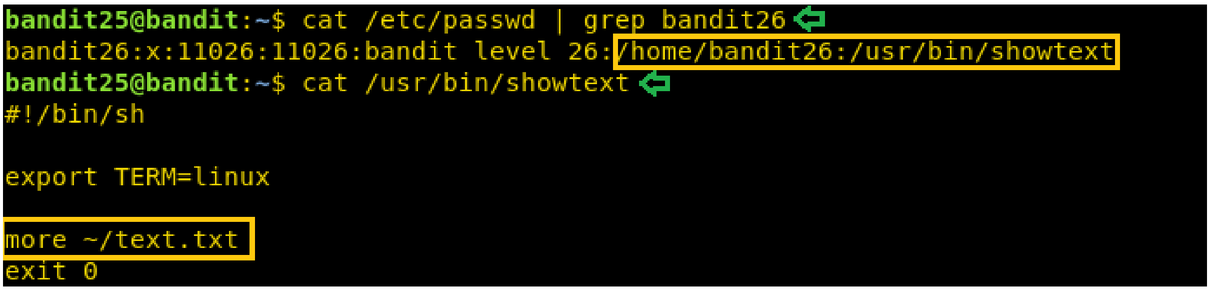
We saw that a session was generated but it displayed a pattern as below and then the session was closed.



```
Enjoy your stay!

Connection to localhost closed.
```

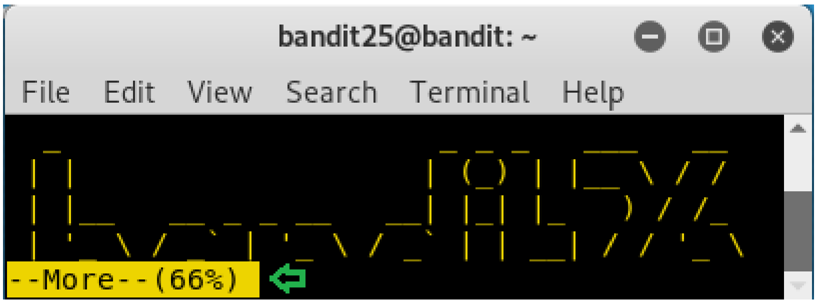
After a bit enumeration, here and there. It hit us to check the /etc/passwd file. As this was a machine with lots of users so we used the grep command to get a refined result for the bandit26 user. It gave us a file called showtext. We read the file showtext using the cat command. It shows us that 'more' is used with the text file that shows us the pattern we saw before. Now, this gave us an idea that we need to provoke the more command. To do this we will have to decrease the size of the terminal so that it can't display that pattern.

```
cat /etc/passwd | grep bandit26
cat /usr/bin/showtext
```



```
bandit25@bandit:~$ cat /etc/passwd | grep bandit26
bandit26:x:11026:11026:bandit level 26:/home/bandit26:/usr/bin/showtext
bandit25@bandit:~$ cat /usr/bin/showtext
#!/bin/sh

export TERM=linux

more ~/text.txt
exit 0
```

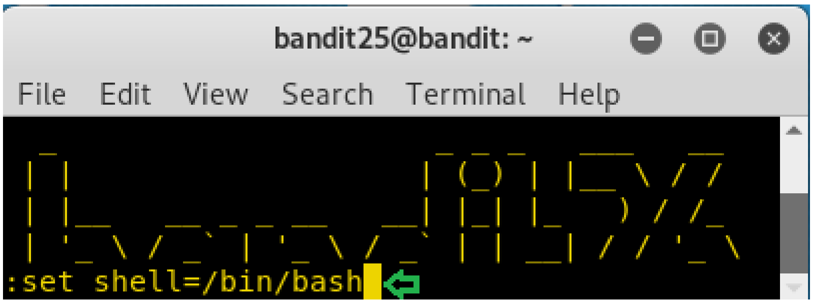
So, we decreased the size of the terminal as shown in the image and then again tried to login. This will trigger the 'more'. Now press 'v' to enable vi editor.



```
--More--(66%)
```

Now, we will write the following command to invoke a shell here as shown in the given image.

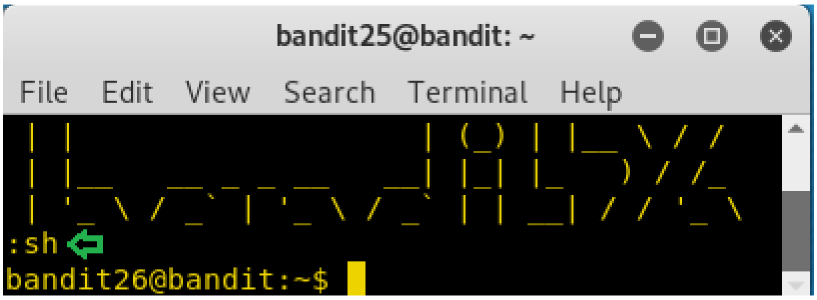
```
:set shell=/bin/bash
```



```
:set shell=/bin/bash
```

As we can see in the given image that we have the shell for bandit26.

```
:sh
```

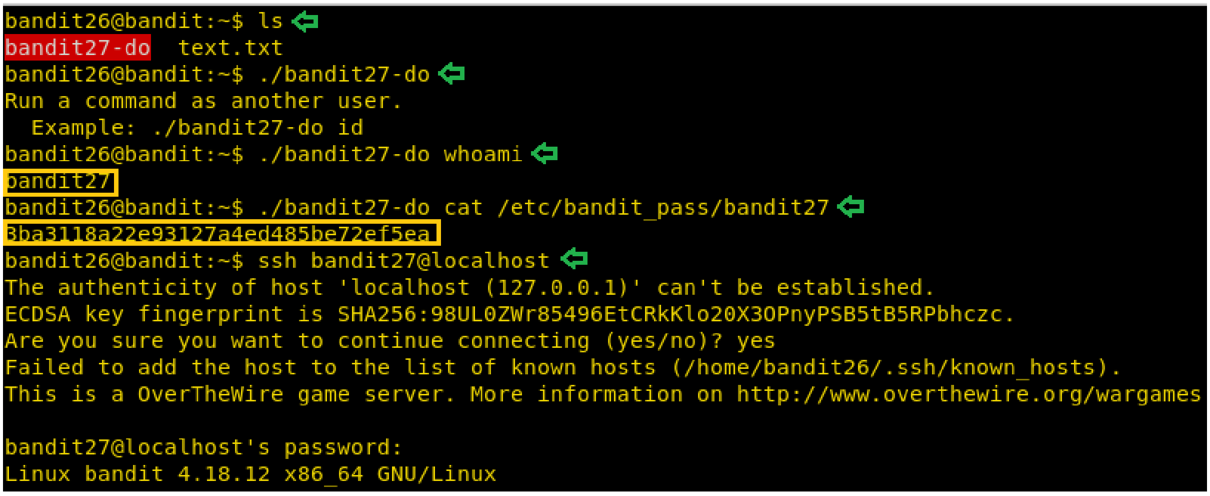


```
:sh
bandit26@bandit:~$
```

Level 26-27

On this level, we are not given any hints. We are on our own on this. So, we like to see what we have to work upon in the current directory. We ran ls command to find a script bandit27-do. Let's execute the script to see if we get any message or hint. It does one better, it gives us an example. This script basically runs the command it is given as user bandit27. So now that we can run commands as user bandit27. Let's read the password file located at /etc/bandit_pass/bandit27. Now that we have the password for the next level, we will login as bandit27 using SSH.

```
ls
./bandit27-do
./bandit27-do whoami
./bandit27-do cat /etc/bandit_pass/bandit27
ssh bandit27@localhost
```

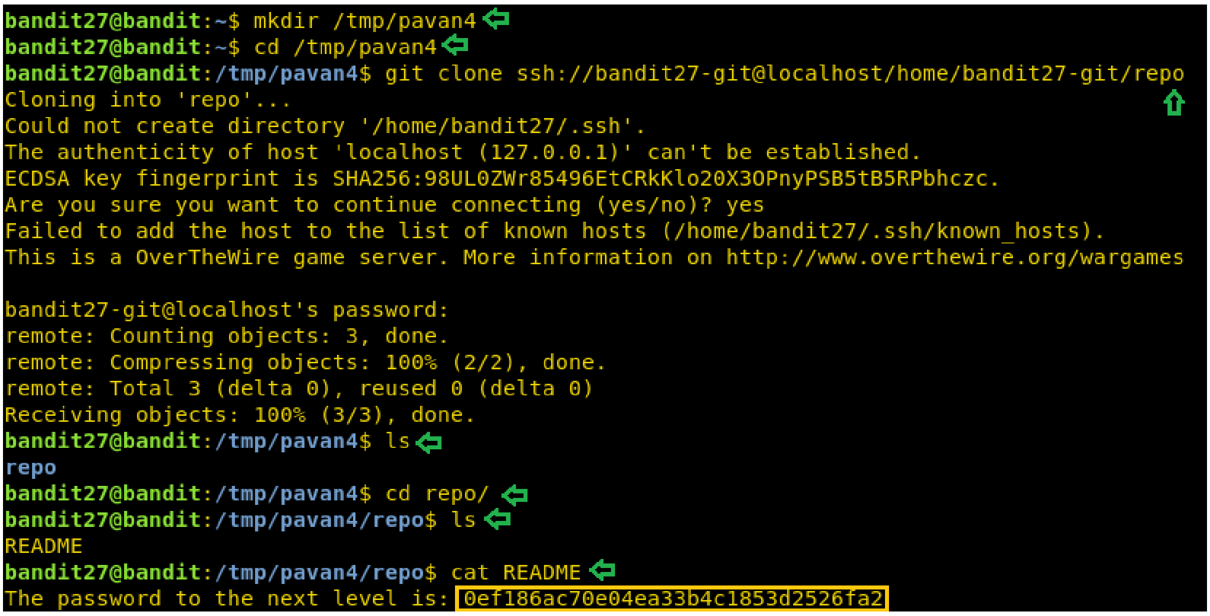


```
bandit26@bandit:~$ ls
bandit27-do  text.txt
bandit26@bandit:~$ ./bandit27-do
Run a command as another user.
  Example: ./bandit27-do id
bandit26@bandit:~$ ./bandit27-do whoami
bandit27
bandit26@bandit:~$ ./bandit27-do cat /etc/bandit_pass/bandit27
3ba3118a22e93127a4ed485be72ef5ea
bandit26@bandit:~$ ssh bandit27@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)? yes
Failed to add the host to the list of known hosts (/home/bandit26/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit27@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 27-28

On this level, we are informed that there is a git repository and the password for that repository is the same password that was used to login in as user bandit27. We are required to clone the repository. Now we need to have the write permission to clone a repository. So, we create a directory in the tmp directory. After cloning let's list all the file in the repo. We find a README file. Upon reading that file we get the password for the next level.

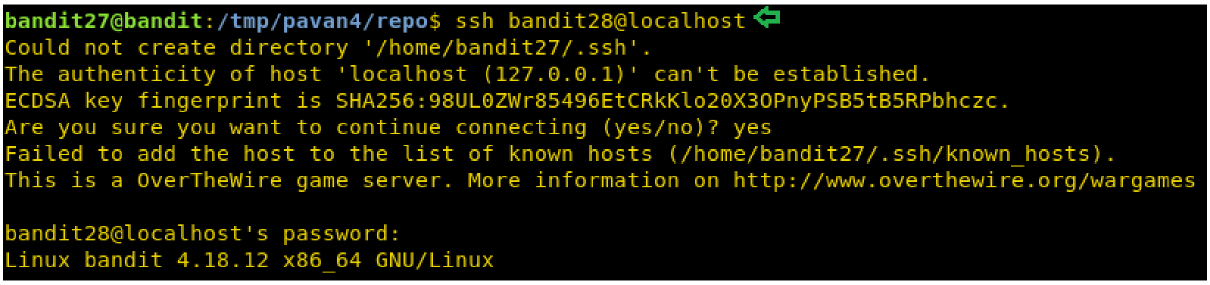
```
mkdir /tmp/pavan4
cd /tmp/pavan4
git clone ssh://bandit27-git@localhost/home/bandit27-git/repo
ls
cd repo
ls
cat README
```



```
bandit27@bandit:~$ mkdir /tmp/pavan4
bandit27@bandit:~$ cd /tmp/pavan4
bandit27@bandit:/tmp/pavan4$ git clone ssh://bandit27-git@localhost/home/bandit27-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 SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
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_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/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/pavan4$ ls
repo
bandit27@bandit:/tmp/pavan4$ cd repo/
bandit27@bandit:/tmp/pavan4/repo$ ls
README
bandit27@bandit:/tmp/pavan4/repo$ cat README
The password to the next level is: 0ef186ac70e04ea33b4c1853d2526fa2
```

Now that we have the password for the next level, we will login as bandit28 using SSH.

```
ssh bandit28@localhost
```

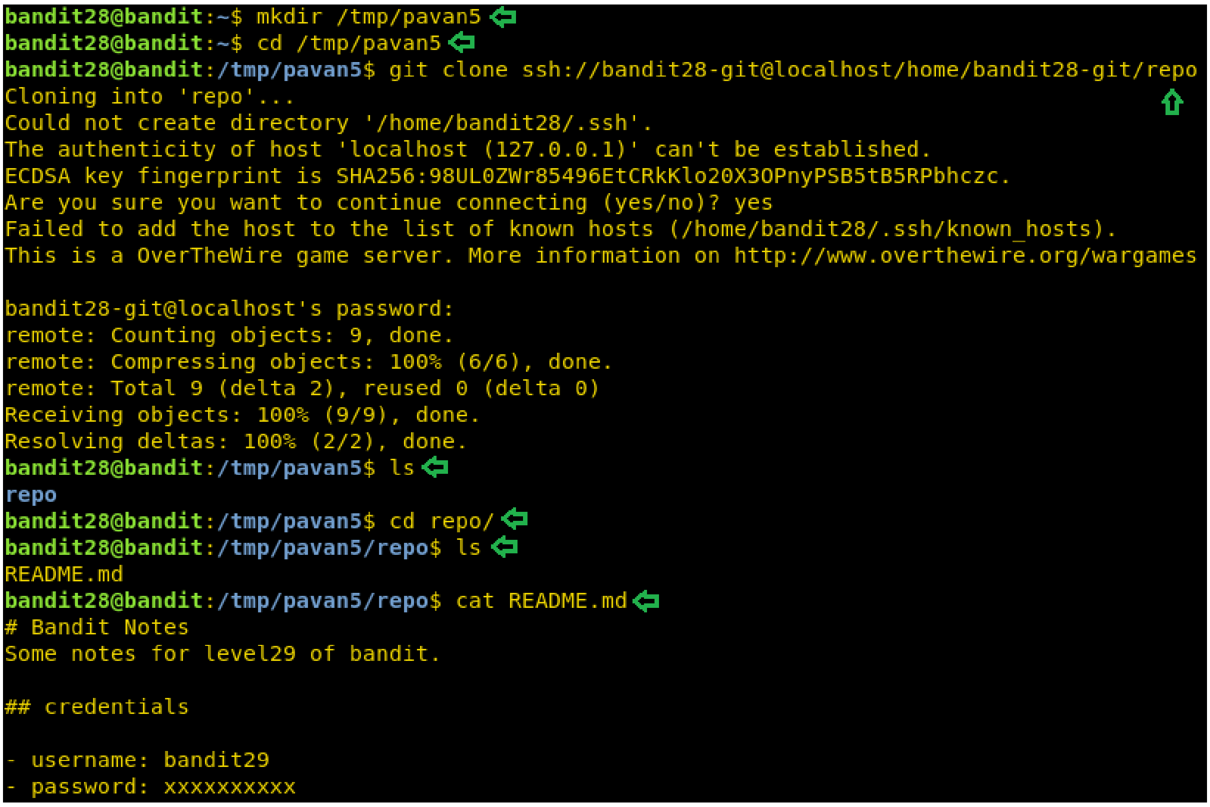


```
bandit27@bandit:/tmp/pavan4/repo$ ssh bandit28@localhost
Could not create directory '/home/bandit27/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X30PnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit27/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit28@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 28-29

On this level, we are informed that there is a git repository and the password for that repository is the same password that was used to login in as user bandit28. We are required to clone the repository. Now we need to have the write permission to clone a repository. So, we create a directory in the tmp directory. After cloning let’s list all the file in the repo. We find a README file. Upon reading that file we see that password is hidden.

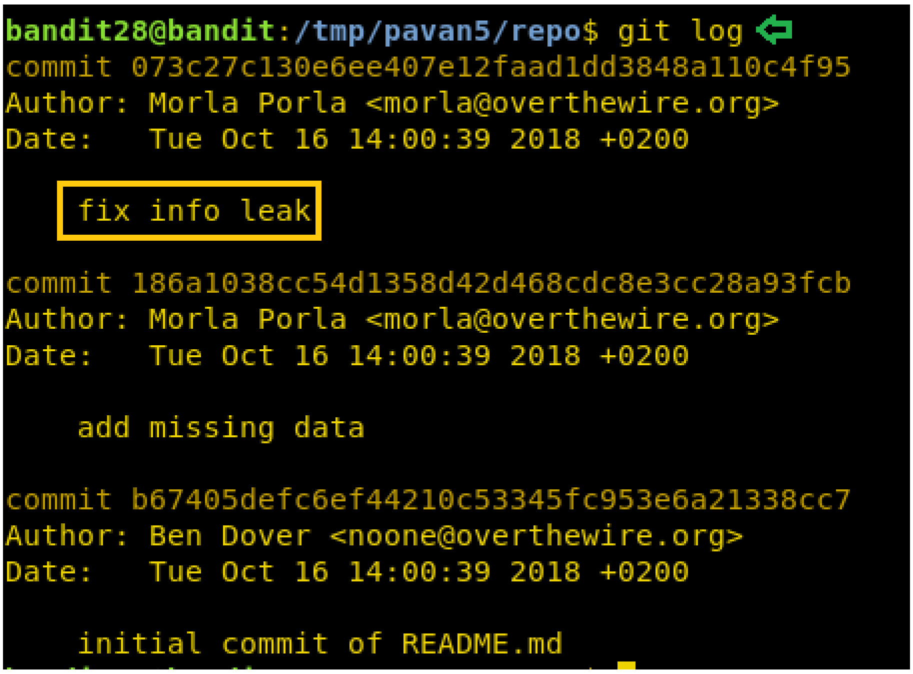
```
mkdir /tmp/pavan5
cd /tmp/pavan5
git clone ssh://bandit28-git@localhost/home/bandit28-git/repo
ls
cd repo/
ls
cat README.md
```



```
bandit28@bandit:~$ mkdir /tmp/pavan5
bandit28@bandit:~$ cd /tmp/pavan5
bandit28@bandit:/tmp/pavan5$ git clone ssh://bandit28-git@localhost/home/bandit28-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_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

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/pavan5$ ls
repo
bandit28@bandit:/tmp/pavan5$ cd repo/
bandit28@bandit:/tmp/pavan5/repo$ ls
README.md
bandit28@bandit:/tmp/pavan5/repo$ cat README.md
# Bandit Notes
Some notes for level29 of bandit.

## credentials

- username: bandit29
- password: xxxxxxxxxx
```

Maybe the password was inside the file but was removed. Good thing is that whenever a change is made in a git, a log entry is created. Let's check that log, we can see that the author of git has made the latest commit named 'fix info leak'. We need to check out this commit.

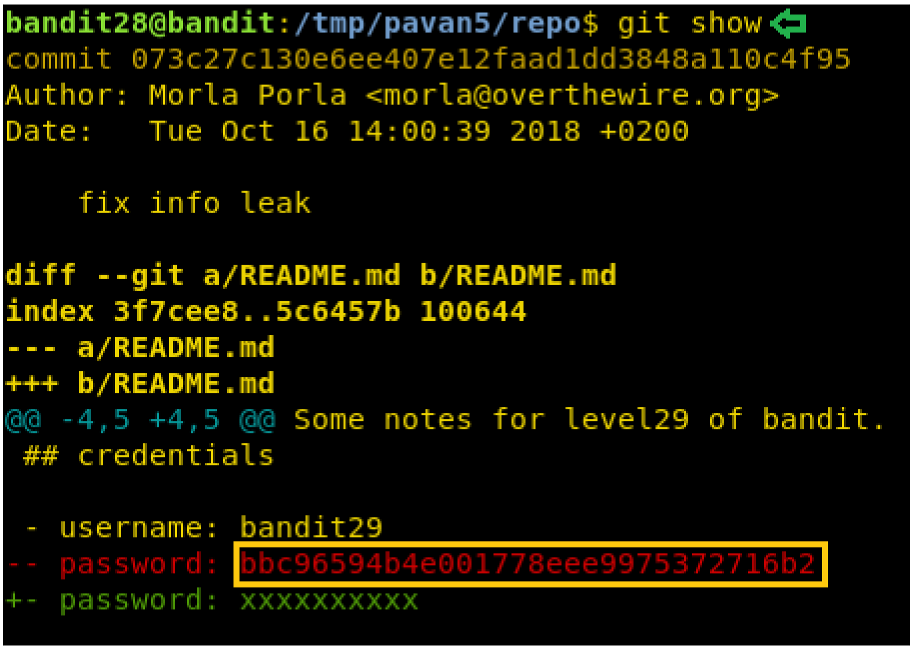
```
git log
```



```
bandit28@bandit:/tmp/pavan5/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
```

To see the changes made in the commit, we will use the git show command to read the changes made. As expected, we found the password inside this commit.

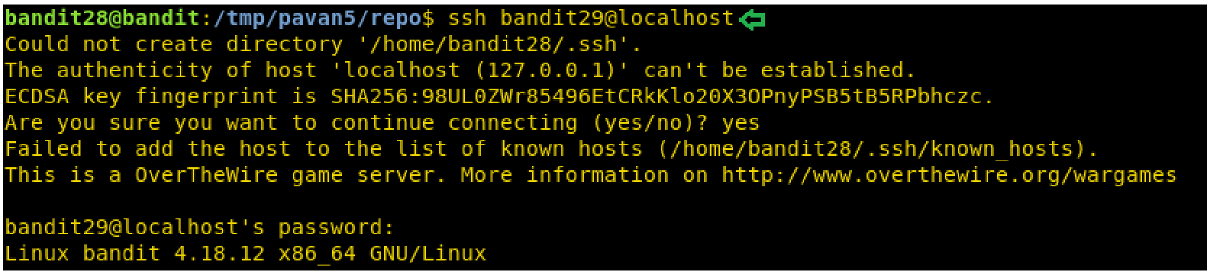
```
git show 073c27c130e6ee407e12faad1dd3848a110c4f95
```



```
bandit28@bandit:/tmp/pavan5/repo$ git show
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
```

Now that we have the password for the next level, we will login as bandit29 using SSH.

```
ssh bandit29@localhost
```

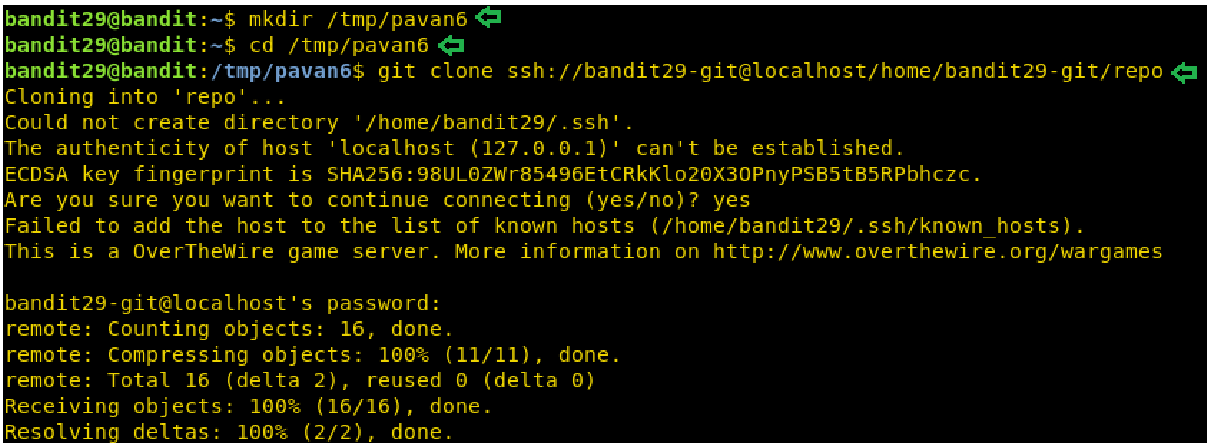


```
bandit28@bandit:/tmp/pavan5/repo$ ssh bandit29@localhost
Could not create directory '/home/bandit28/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit28/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit29@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 29-30

On this level, we are informed that there is a git repository and the password for that repository is the same password that was used to login in as user bandit29. We are required to clone the repository. Now we need to have the write permission to clone a repository. So, we create a directory in the tmp directory. Now we will clone the repository inside this directory.

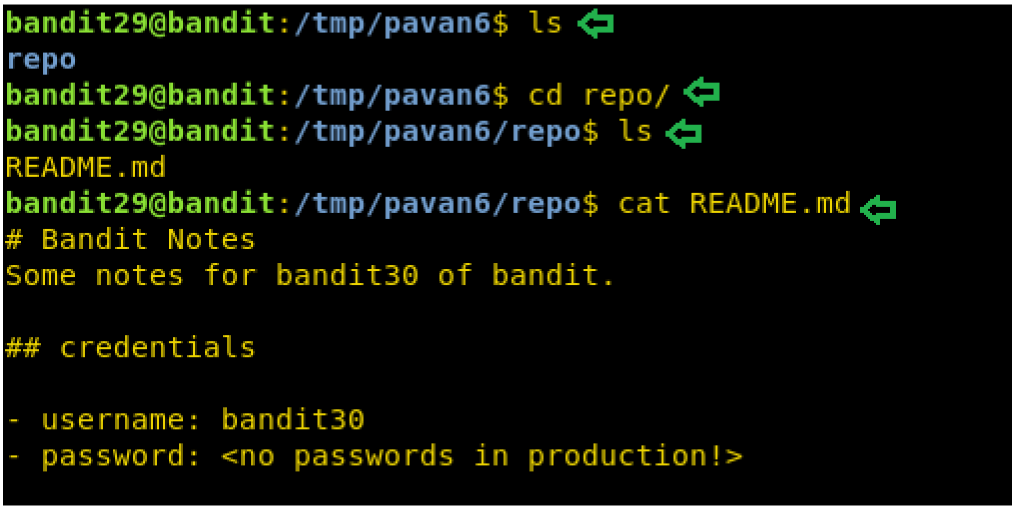
```
mkdir /tmp/pavan6
cd /tmp/pavan6
git clone ssh://bandit29-git@localhost/home/bandit29-git/repo
```



```
bandit29@bandit:~$ mkdir /tmp/pavan6
bandit29@bandit:~$ cd /tmp/pavan6
bandit29@bandit:/tmp/pavan6$ git clone ssh://bandit29-git@localhost/home/bandit29-git/repo
Cloning into 'repo'...
Could not create directory '/home/bandit29/.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/bandit29/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit29-git@localhost's password:
remote: Counting objects: 16, done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 16 (delta 2), reused 0 (delta 0)
Receiving objects: 100% (16/16), done.
Resolving deltas: 100% (2/2), done.
```

After cloning let's list all the file in the repo. We find a README file. Here we are told that there is no password in production. Now its time to enumerate this git.

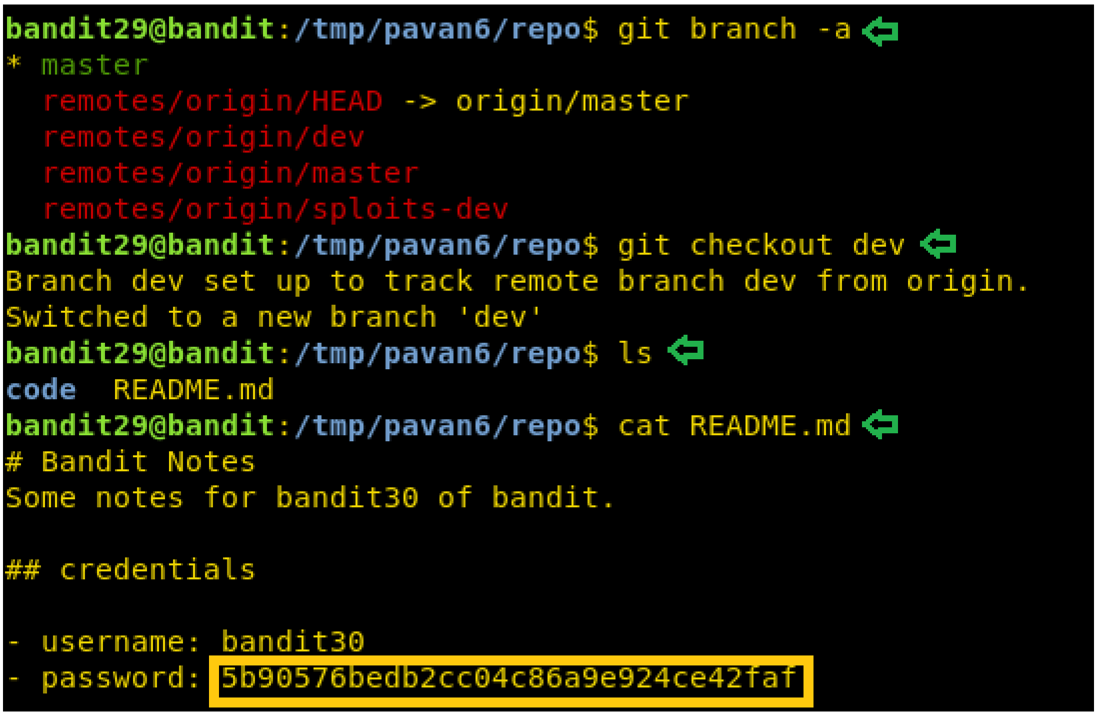
```
ls
cd repo/
ls
cat README.md
```



```
bandit29@bandit:/tmp/pavan6$ ls
repo
bandit29@bandit:/tmp/pavan6$ cd repo/
bandit29@bandit:/tmp/pavan6/repo$ ls
README.md
bandit29@bandit:/tmp/pavan6/repo$ cat README.md
# Bandit Notes
Some notes for bandit30 of bandit.

## credentials

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

We list all the branches in this git using the git branch command. It shows us that we. have another branch called dev. Let’s check out this branch for the password. After switching to this branch, we run ls command to see that we have a README file. Upon reading that file we get the credentials.

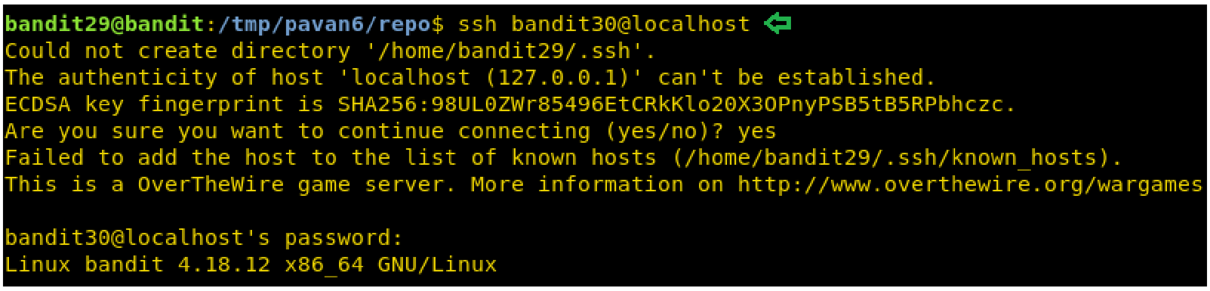
```
git branch -a
git checkout dev
cat README.md
```



```
bandit29@bandit:/tmp/pavan6/repo$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/dev
  remotes/origin/master
  remotes/origin/sploits-dev
bandit29@bandit:/tmp/pavan6/repo$ git checkout dev
Branch dev set up to track remote branch dev from origin.
Switched to a new branch 'dev'
bandit29@bandit:/tmp/pavan6/repo$ ls
code  README.md
bandit29@bandit:/tmp/pavan6/repo$ cat README.md
# Bandit Notes
Some notes for bandit30 of bandit.

## credentials

- username: bandit30
- password: 5b90576bedb2cc04c86a9e924ce42faf
```

Now that we have the password for the next level, we will login as bandit30 using SSH.

```
ssh bandit30@localhost
```

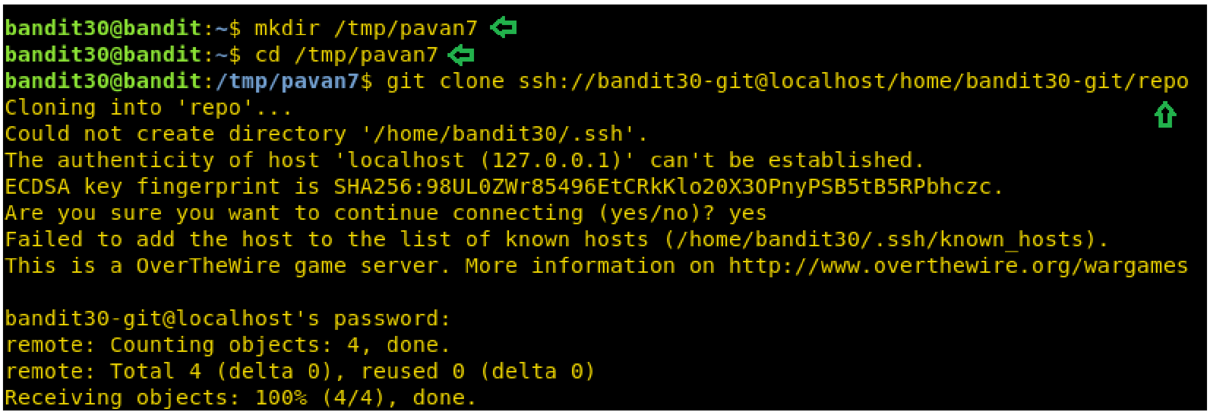


```
bandit29@bandit:/tmp/pavan6/repo$ ssh bandit30@localhost
Could not create directory '/home/bandit29/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit29/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit30@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 30-31

On this level, we are informed that there is a git repository and the password for that repository is the same password that was used to login in as user bandit30. We are required to clone the repository. Now we need to have the write permission to clone a repository. So, we create a directory in the tmp directory. Now we will clone the repository inside this directory.

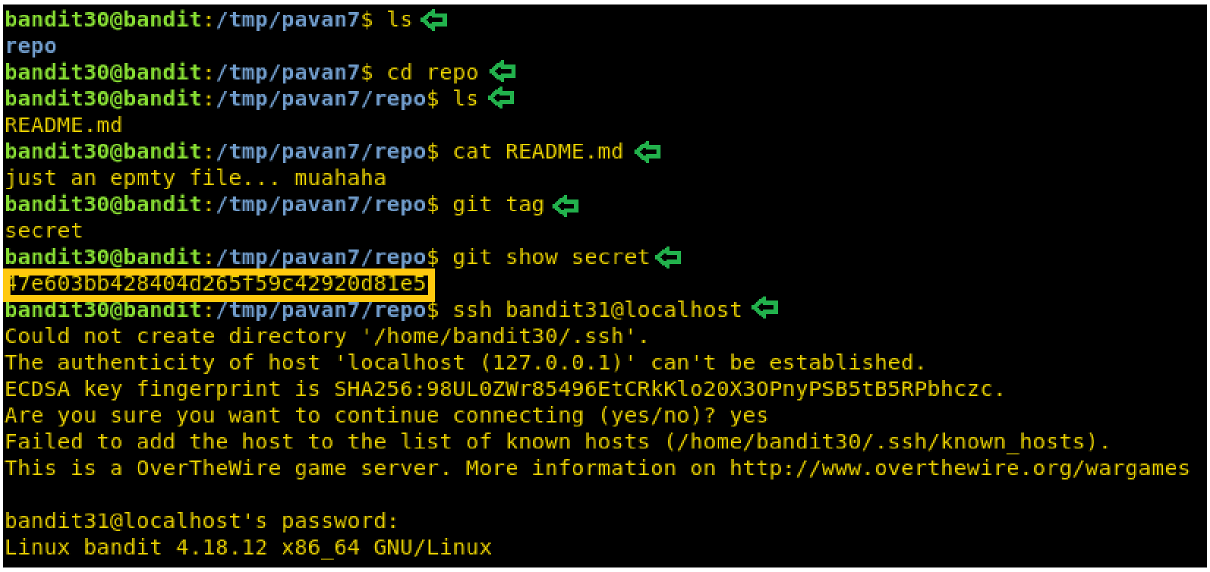
```
mkdir /tmp/pavan7

cd /tmp/pavan7

git clone ssh://bandit30-git@localhost/home/bandit30-git/repo
```



```
bandit30@bandit:~$ mkdir /tmp/pavan7
bandit30@bandit:~$ cd /tmp/pavan7
bandit30@bandit:/tmp/pavan7$ git clone ssh://bandit30-git@localhost/home/bandit30-git/repo
Cloning into 'repo'...
Could not create directory '/home/bandit30/.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/bandit30/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit30-git@localhost's password:
remote: Counting objects: 4, done.
remote: Total 4 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (4/4), done.
```

After cloning let’s list all the file in the repo. We find a README file. Here we are told that it is an empty file. Now it’s time to enumerate this git. Git has the ability to tag specific points in a repository’s history as being important. We can enumerate that tag. On looking carefully, we find the tag secret. On reading that tag we find the password we were looking for on this level. Now that we have the password for the next level, we will login as bandit31 using SSH.

```
ls
cd repo
ls
cat README.md
git tag
git show secret
ssh bandit31@localhost
```

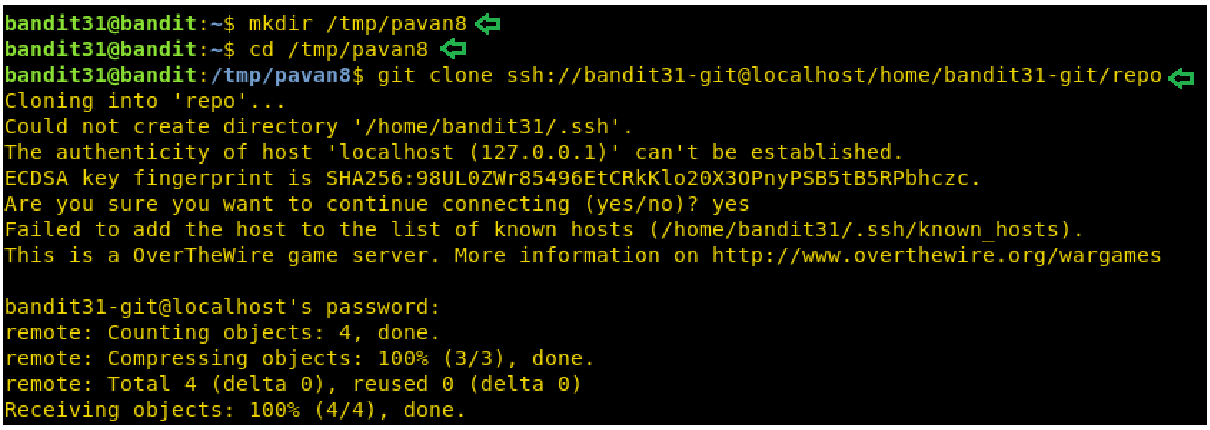


```
bandit30@bandit:/tmp/pavan7$ ls
repo
bandit30@bandit:/tmp/pavan7$ cd repo
bandit30@bandit:/tmp/pavan7/repo$ ls
README.md
bandit30@bandit:/tmp/pavan7/repo$ cat README.md
just an epmty file... muahaha
bandit30@bandit:/tmp/pavan7/repo$ git tag
secret
bandit30@bandit:/tmp/pavan7/repo$ git show secret
47e603bb428404d265f59c42920d81e5
bandit30@bandit:/tmp/pavan7/repo$ ssh bandit31@localhost
Could not create directory '/home/bandit30/.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/bandit30/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit31@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 31-32

On this level, we are informed that there is a git repository and the password for that repository is the same password that was used to login in as user bandit31. We are required to clone the repository. Now we need to have the write permission to clone a repository. So, we create a directory in the tmp directory. Now we will clone the repository inside this directory.

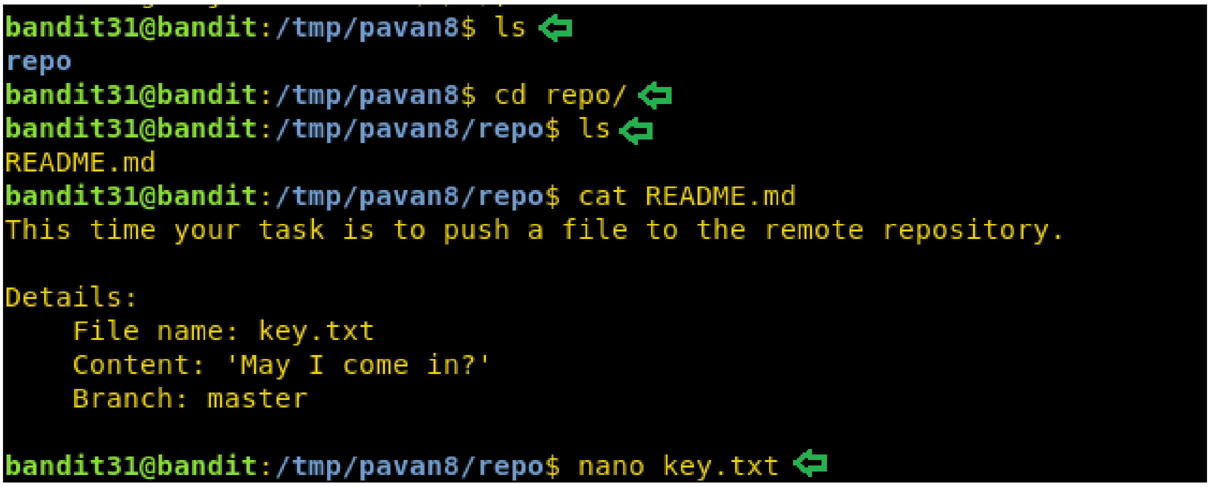
```
mkdir /tmp/pavan8
cd /tmp/pavan8
git clone ssh://bandit31-git@localhost/home/bandit31-git/repo
```



```
bandit31@bandit:~$ mkdir /tmp/pavan8
bandit31@bandit:~$ cd /tmp/pavan8
bandit31@bandit:/tmp/pavan8$ git clone ssh://bandit31-git@localhost/home/bandit31-git/repo
Cloning into 'repo'...
Could not create directory '/home/bandit31/.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/bandit31/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit31-git@localhost's password:
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (4/4), done.
```

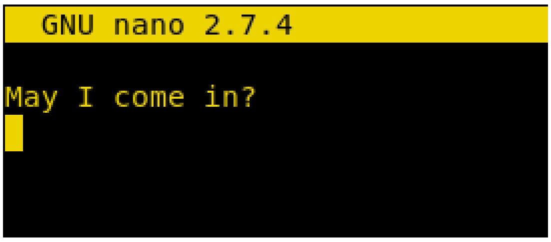
After cloning let’s list all the file in the repo. We find a README file. Here we are told that in order to get the password for the next level, we have to push a file in the remote repository. This file must be named key.txt and should contain the content May I come in?

```
ls
cd repo/
ls
cat README.md
nano key.txt
```



```
bandit31@bandit:/tmp/pavan8$ ls
repo
bandit31@bandit:/tmp/pavan8$ cd repo/
bandit31@bandit:/tmp/pavan8/repo$ ls
README.md
bandit31@bandit:/tmp/pavan8/repo$ cat README.md
This time your task is to push a file to the remote repository.

Details:
    File name: key.txt
    Content: 'May I come in?'
    Branch: master

bandit31@bandit:/tmp/pavan8/repo$ nano key.txt
```

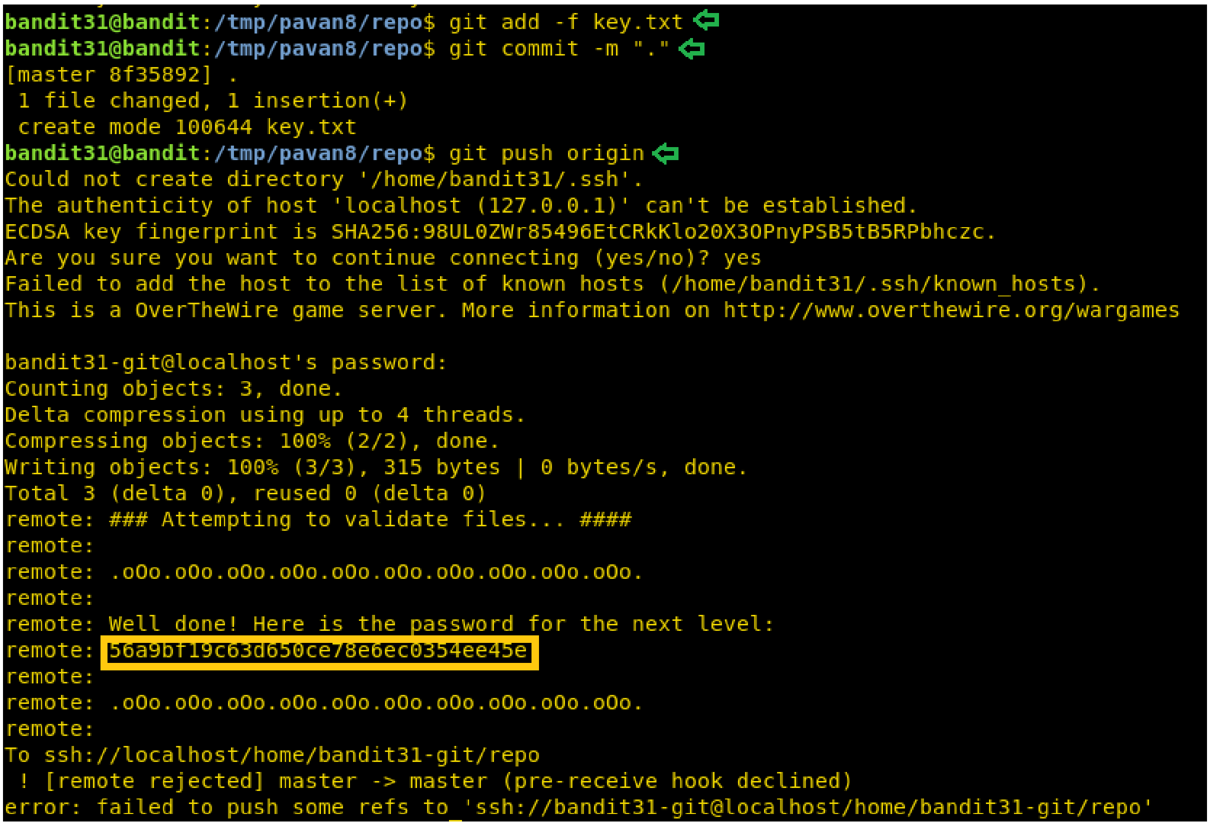
So, we create a text file name key using nano and enter the phrase “May I come in?” in it.



```
  GNU nano 2.7.4

May I come in?
```

Now we add the file to the repository and commit to that entry. And finally, push it into the origin branch. This step requires the password for the current user. As we can see in the given image that we have the password for the next level.

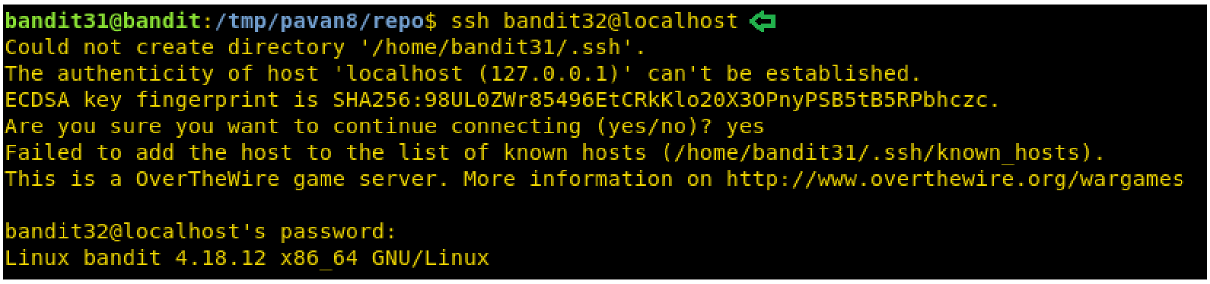
```
git add -f key.txt
git commit -m "."
git push origin
```



```
bandit31@bandit:/tmp/pavan8/repo$ git add -f key.txt
bandit31@bandit:/tmp/pavan8/repo$ git commit -m "."
[master 8f35892] .
 1 file changed, 1 insertion(+)
 create mode 100644 key.txt
bandit31@bandit:/tmp/pavan8/repo$ git push origin
Could not create directory '/home/bandit31/.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/bandit31/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit31-git@localhost's password:
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 315 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote: ### Attempting to validate files... ####
remote:
remote: .oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.
remote:
remote: Well done! Here is the password for the next level:
remote: 56a9bf19c63d650ce78e6ec0354ee45e
remote:
remote: .oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.oOo.
remote:
To ssh://localhost/home/bandit31-git/repo
 ! [remote rejected] master -> master (pre-receive hook declined)
error: failed to push some refs to 'ssh://bandit31-git@localhost/home/bandit31-git/repo'
```

Now that we have the password for the next level, we will login as bandit32 using SSH.

```
ssh bandit32@localhost
```

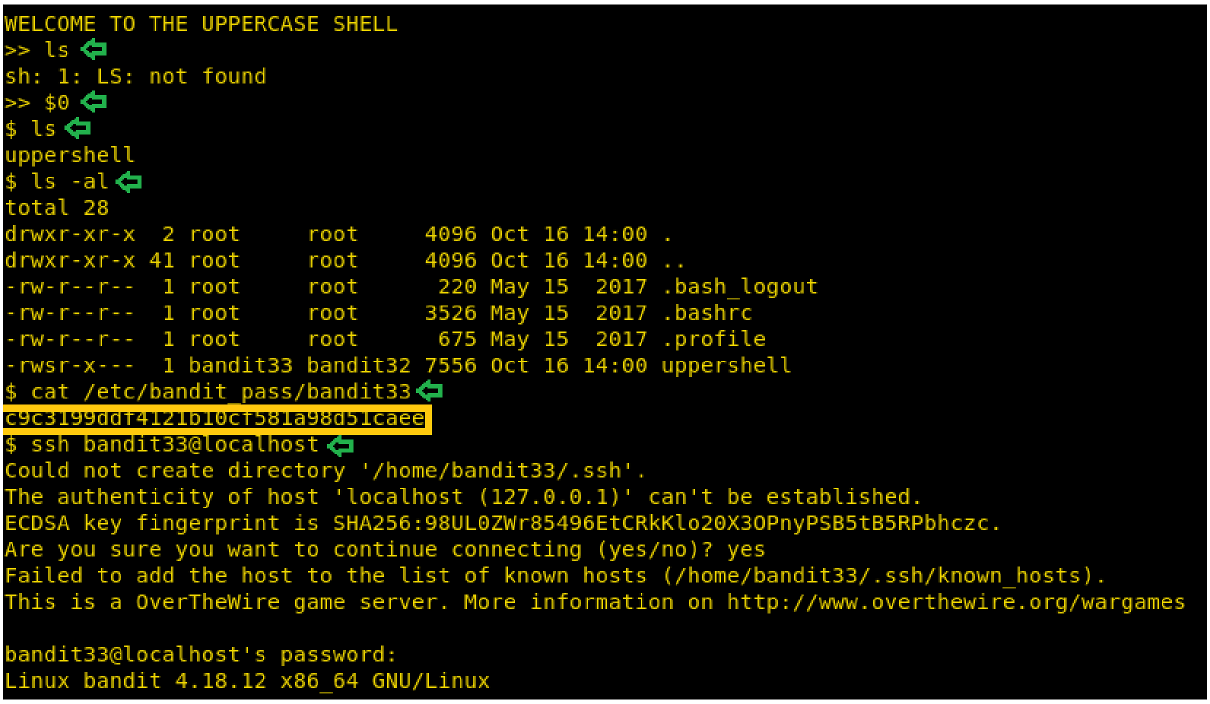


```
bandit31@bandit:/tmp/pavan8/repo$ ssh bandit32@localhost
Could not create directory '/home/bandit31/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/home/bandit31/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit32@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 32-33

On reaching this level, we are greeted with a message “Welcome to the Uppercase shell”. To understand what it does, we ran ls command but we got an error. On close inspection of the error message, we understand that it states that the LS command is not found. It means that the shell converts my commands to Uppercase before executing. For this level, we are given a hint “it’s time for another escape”. This made us curious about escape characters. Upon brief research, we found that we can bypass this uppercase shell using an escape character ‘$0’. We were right. We got the bash. Let’s list all files using ls -al command. We see that the owner of uppercase is bandit33. So, we can access the /etc/bandit_pass/bandit33 file to get the password for the next level. After getting the password, we will login as bandit33 using SSH.

```
ls
$0
ls -al
cat /etc/bandit_pass/bandit33
ssh bandit33@localhost
```

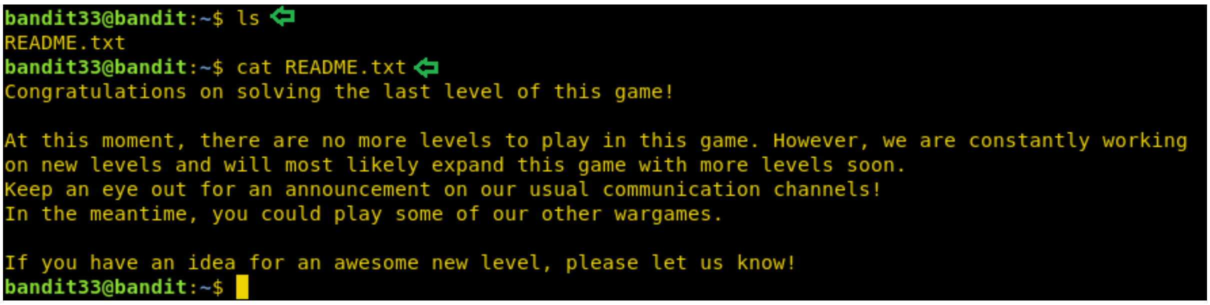


```
WELCOME TO THE UPPERCASE SHELL
>> ls
sh: 1: LS: not found
>> $0
$ ls
uppershell
$ ls -al
total 28
drwxr-xr-x  2 root     root     4096 Oct 16 14:00 .
drwxr-xr-x 41 root     root     4096 Oct 16 14:00 ..
-rw-r--r--  1 root     root      220 May 15  2017 .bash_logout
-rw-r--r--  1 root     root     3526 May 15  2017 .bashrc
-rw-r--r--  1 root     root      675 May 15  2017 .profile
-rwsr-x---  1 bandit33 bandit32 7556 Oct 16 14:00 uppershell
$ cat /etc/bandit_pass/bandit33
c9c3199ddf4121b10cf581a98d51caee
$ ssh bandit33@localhost
Could not create directory '/home/bandit33/.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/bandit33/.ssh/known_hosts).
This is a OverTheWire game server. More information on http://www.overthewire.org/wargames

bandit33@localhost's password:
Linux bandit 4.18.12 x86_64 GNU/Linux
```

Level 33

This is the final level for now as the bandit team is working on creating more levels. We connected to this level as use bandit33. After connecting we run ls command to see the list of files we have in the current directory. We see that we have a README file. On opening that file, we see the final flag and a brief message from the Over the Wire Team. This concludes this series for now. We will solve more levels as soon as Over the Wire team publishes more levels.

```
ls
cat README.txt
```



```
bandit33@bandit:~$ ls
README.txt
bandit33@bandit:~$ cat README.txt
Congratulations on solving the last level of this game!

At this moment, there are no more levels to play in this game. However, we are constantly working
on new levels and will most likely expand this game with more levels soon.
Keep an eye out for an announcement on our usual communication channels!
In the meantime, you could play some of our other wargames.

If you have an idea for an awesome new level, please let us know!
bandit33@bandit:~$
```

Conclusion

Hence, one can make use of these commands as a cybersecurity professional to assess vulnerabilities on systems and keep these systems away from threat.

References

- https://www.hackingarticles.in/overthewire-bandit-walkthrough-1-14/
- https://www.hackingarticles.in/overthewire-bandit-walkthrough-14-21/
- https://www.hackingarticles.in/overthewire-bandit-walkthrough-21-34/
- https://overthewire.org/wargames/bandit/