Over The Wire -- Bandit

By Martin Ren

Hint

Login

Since the port of server changed recently, people need to choose port while login. If you just use

ssh bandit0@bandit.labs.overthewire.org

It will use default port which is port 20. So you need to add **-p 2220** to choose port.

Level 0 - Level 1

From hint, it says password is stored in a file called readme in home directory which is the directory when you log in. **cat** is one of the most frequently used command which can display contents of file on terminal. So **cat readme** can display password from that file.

Level 1 - Level 2

When you type **cat -** on terminal, it doesn't show password but wait for your input. Because when cat read the string - as a filename, - is treated as synonym for stdin. One way to read the file is you can prefix the filename with a path **./-** where ./ means current directory.

Level 2 - Level 3

If you type **cat spaces in this file**, it will recognise as read four files. Space in terminal is used to split command or options. **cat spaces\ in\ this\ file** can transfer to normal space in a filename. Also, **tab** can help you implement filename automatically. E.g. **cat s** then press **tab**

Level 3 - Level 4

Using **ls** command can list directory contents. Using **cd *directory*** can change directory. There is no file when you use **ls** in inhere directory. Password is in hidden file as the hint mentioned. **ls -a** can list all files in current directory which include hidden files.

Level 4 - Level 5

One way to solve this level is open every file in inhere directory and find password. The other way is use **egrep** to read all files and match password. egrep is a command search for a pattern using extended regular expressions. It is same as running grep with -E option. Because only one file is human readable and contain password. **egrep "^[a-zA-Z0-9]+$" inhere/*** can match content has letters and numbers.

Level 5 - Level 6

ls -l can help you get information of file such as size. **ls -laR** can list information of all files in subdirectories recursively. Use **egrep | "1033"** can match file that contain password.

Level 6 - Level 7

find command can help you search file. **-user *username*** can set user you want to find. **-group *group_name*** can find groups. **-size *size_of_file*** can find size that you want.

Level 7 - Level 8

If we read file, we can know that each line contain line number and a word. So we need get number of line has word "millionth" and find next line. **-n** option in cat means number of output lines. Use egrep to match word "millionth" and get line number. Then use egrep to match next line of word "millionth" which is password.

Level 8 - Level 9

At first, we can use **sort** to sort duplicate data together. Then use **uniq -u** which only print unique lines. Note: | can split multiple command in one line. For example, **cat data.txt | sort | uniq -u**

Level 9 - Level 10

There are lots of messy message if we read file. We can use egrep to match string only contain "=". If you do it, you'll find there are lots of results. Another hint from website is **several "=" characters** which means we can use grep to match string contain more than one "=" characters.

Level 10 - Level 11

Use **base64 -d** can decode content of file when you **cat** it.

Level 11 - Level 12

tr is used to substitute or delete copied input. For example, **cat file | tr "abc" "ABC"** means change all lower letter a, b or c to upper letter. Letters in file has been rotated by 13 positions in this level which means "a" change to "n", "b" change to "o".

Level 12 - Level 13

xxd is a command to create hex dump of a given file or input. In this level, we need convert hex dump convert to binary form which use **-r** option. Also we need to name an output file to store. **file** command is used to determine file type. **zcat** can decompressed file that compressed use gzip. **bzcat** can decompress bzip2 file. And **tar xO** can decompress tar file.

Level 13 - Level 14

i option in ssh can select file which the identity for public key authentication is read. In this level the file is in home directory called sshkey.private

Level 14 - Level 15

If you do not know how internet work, you can read materials they provide. **nc** can arbitrary TCP/UDP connections. **nc 127.0.0.1 3000** means connect to port 3000 on localhost where 127.0.0.1 is address of localhost.

Level 15 - Level 16

It is hard to explain how to solve this level. Using **echo "BfMYroe26WYalil77FoDi9qh59eK5xNr" | openssl s_client -connect localhost:30001 -ign_eof** can get password.

Level 16 - Level 17

netstat can print network connections. **listen** option in **netstat** only print ports that are listening. Similar to level 15 - level 16, but we need to check every port to find correct one. Then we use private key that we get to log into next level.

Level 17 - Level 18

This level is relatively easy. **diff** command can compare and print differences between two files.

Level 18 - Level 19

.bashrc file run a script that let use logout when they log in. So we need to bypass **.bashrc** file when we log in. When you use **ssh** to log in, add **/bin/dash** at the end which will not run **.bashrc** automatically. Then you can **cat** file contained password.

Level 19 - Level 20

Execute setuid binary in home directory and use it to cat password from **/etc/bandit_pass/bandit20**
Note: You can run program with following command ***./program_name***

Level 20 - Level 21

Similar to previous level, but you need two terminal windows. One run the setuid command and set network daemon, and the other one listen from the first terminal. When you send password of this level from first terminal, you can get password for this level.

Level 21 - Level 22

Using **cd** go to **/etc/cron.d/** and you can get list of scripts. Using **cat** to read script for this level which is **cronjob_bandit22**. You can get that there is shell script called **cronjob_bandit22.sh** in **/usr/bin/**. In cronjob_bandit22.sh, **cat /etc/bandit_pass/bandit22 > /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv** means get content from first file and write into second file. So we can cat second file to get password because we have no permission to read first file.

Level 22 - Level 23

Similar to previous level but more difficult. You can not run shell script they provide. But you can run code in your terminal. So at first, use **whoami** to get user name. You should use username bandit23 because you want to get password of level 23. Then **"I am user bandit23" | md5sum | cut -d ' ' -f 1** can get filename that contain password.

Level 23 - Level 24

Go to **/var/spool/bandit24** and you can write your own script and run it. But the other script will delete your work after you run it. We just need to write a simple script as in level 21. Just copy password to other file. **nano** or **vi** can help you to edit your script. Notice that before you run your code, you need use **chmod** to change permission of files. You cannot run your script if you forget to change permission.

Level 24 - Level 25

We need a script to help using brute force to send different security code and get password. We create a script in **/var/spool/bandit24** which is same position as previous level.

```
#!/bin/bash
a=0
while [ $a -lt 10000 ] do
        echo "UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ `printf %04d $a`" | nc 127.0.0.1 30002
        >> /tmp/bandit25_pass.txt &
        a=`expr $a + 1`
done
```

If you never use shell before, this code can help you get password. **&** means send next request without waiting for answer which can get password faster. Then use **egrep** to get password where I taught at level 4 to level 5.

Level 25 - Level 26

This one is tricky and hard for me. **cat /etc/passwd** you can get list of shell for each level. You can find shell for bandit26 are different with others. **cat /usr/bin/showtext** you can get code write for level 26.

Tricky thing is, you need familiar with **more** command and **vi** editor. First, you need change window small enough which can show around 5 lines. Then log into level 26 by using private key in home directory. Enter **vi** to edit file. **:r /etc/bandit_pass/bandit26** can get password after you finish editting.

Answer

Level0 - Level1

```
cat readme
boJ9jbbUNNfktd78OOpsqOltutMc3MY1
```

Level1 - Level2

```
cat ./-
CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9
```

Level2 - Level3

```
cat spaces\ in\ this\ filename
UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK
```

Level3 - Level4

```
cat inhere/.hidden
pIwrPrtPN36QITSp3EQaw936yaFoFgAB
```

Level4 - Level5

```
egrep "^[a-zA-Z0-9]+$" ./*
koReBOKuIDDepwhWk7jZC0RTdopnAYKh
```

Level5 - Level6

```
cd inhere
ls -laR | grep "1033"
ls -laR
(find directory)
cat maybehere07/.file2
DXjZPULLxYr17uwoI01bNLQbtFemEgo7
```

Level6 - Level7

```
cd ../..
find -user bandit7 -group bandit6 -size 33c 2>dev/null cat ./var/lib/dpkg/info/bandit7.password
HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs
```

Level7 - Level8

```
cat -n data.txt | egrep "millionth"
cat -n data.txt | egrep "17525"
cvX2JJa4CFALtqS87jk27qwqGhBM9plV
```

Level8 - Level9

```
cat data.txt | sort | uniq -u
UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR
```

Level9 - Level10

```
cat data.txt | egrep -a "=="+
truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk
```

Level10 - Level11

```
cat data.txt | base64 -d
IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR
```

Level11 - Level12

```
cat data.txt | tr "a-zA-Z" "n-za-mN-ZA-M"
5Te8Y4drgCRfCx8ugdwuEX8KFC6k2EUu
```

Level12 - Level13

```
xxd -r data.txt output.txt
zcat output.txt | bzcat | zcat | tar xO | tar xO | bzcat | tar xO | zcat
8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL
```

Level13 - Level14

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

(Actually we don’t need pass to log into level 14)

```
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
```

Level14 - Level15

```
echo "4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e" | nc 127.0.0.1 30000
BfMYroe26WYalil77FoDi9qh59eK5xNr
```

Level15 - Level16

```
echo “BfMYroe26WYalil77FoDi9qh59eK5xNr” | openssl s_client -connect localhost:30001 -ign_eof
cluFn7wTiGryunymYOu4RcffSxQluehd
```

Level16 - Level17

```
netstat --listen
echo "cluFn7wTiGryunymYOu4RcffSxQluehd" | openssl s_client -connect localhost:31790 -ign_eof
```

(create rsa private key)

```
chmod -400 name
ssh -i key.private bandit17@bandit.labs.overthewire.org
xLYVMN9WE5zQ5vHacb0sZEVqbrp7nBTn
```

Level17 - Level18

```
diff password.new password.old
kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd
```

Level18 - Level19

```
ssh bandit18@bandit.labs.overthewire.org /bin/dash
cat readme
IueksS7Ubh8G3DCwVzrTd8rAVOwq3M5x
```

Level19 - Level20

```
./bandit20-do cat /etc/bandit_pass/bandit20
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
```

Level20 - Level21

```
First terminal: nc -l 30055
Second terminal: ./suconnect 30055
First terminal: GbKksEFF4yrVs6il55v6gwY5aVje5f0j
gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr
```

Level21 - Level22

```
cd /etc/cron.d/
cat cronjob_bandit22
cat /usr/bin/cronjob_bandit22.sh
cat /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI
```

Level22 - Level23

```
cd /etc/cron.d/
cat cronjob_bandit23
cat /usr/bin/cronjob_bandit23.sh
whoami
echo "I am user bandit23" | md5sum | cut -d ' ' -f 1
cat /tmp/8ca319486bfbbc3663ea0fbe81326349
jc1udXuA1tiHqjIsL8yaapX5XIAI6i0n
```

Level23 - Level24

```
cat /etc/cron.d/cronjob_bandit24
cat /usr/bin/cronjob_bandit24.sh
cd /var/spool/bandit24
nano bandit24.sh
        #!/bin/bash
        cat /etc/bandit_pass/bandit24 > /tmp/bandit24_pass.sh
chmod 755 bandit.sh
/usr/bin/cronjob_bandit24.sh
cat /tmp/bandit24_pass.sh
UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ
```

Level24 - Level25

```
cd /tmp/spool/bandit24
nano bandit25.sh
        #!/bin/bash

        a=0

        while [ $a -lt 10000 ]
        do
          echo "UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ `printf %04d $a`" | nc 127.0.0.1
        30002 >> /tmp/bandit25_pass.txt &
          a=`expr $a + 1`
        done
chmod 755 bandit25.sh
./bandit25.sh
cat /tmp/bandit25_pass.txt | egrep [A-Za-z0-9]{32}
uNG9O58gUE7snukf3bvZ0rxhtnjzSGzG
```

Level25 - Level26

```
cat /etc/passwd
cat /usr/bin/showtext
ssh -i bandit26.sshkey bandit26@localhost
v
:r /etc/bandit_pass/bandit26
5czgV9L3Xx8JPOyRbXh6lQbmIOWvPT6Z
```