Bandit Level 1 to 10 (CTF) Over The Wire

Prepared by Syed Abdul Basit Under the supervision of Professor Muhammad Waqar

Assignment : 02

Bandit Level

The goal of this level is for you to log into the game using SSH. The host to which you need to connect is bandit.labs.overthewire.org, on port 2220. The username is bandit0 and the password is bandit0..

In the beginning, you have to :

```
└─$ ssh bandit0@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


                   This is an OverTheWire game server.
          More information on http://www.overthewire.org/w

bandit0@bandit.labs.overthewire.org's password:
```

the password is : bandit0

Bandit Level 0 → Level 1

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

We have to find the password to reach the next level :

```
bandit0@bandit: ~
File  Actions  Edit  View  Help
bandit0@bandit:~$ ls
readme
bandit0@bandit:~$ cat readme
NH2SXQwcBdpmTEzi3bvBHMM9H66vVXjL
bandit0@bandit:~$
```

To connect to the next level password :

Bandit Level 1 → Level 2

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

```
└─$ ssh bandit1@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|

                      This is an OverTheWire game serve
            More information on http://www.overthewire.

bandit1@bandit.labs.overthewire.org's password:
```

To reach the next level:

password : NH2SXQwcBdpmTEzi3bvBHMM9H66vVXjL

— We now have to find the password for the next level :

```
bandit1@bandit: ~
File  Actions  Edit  View  Help
bandit1@bandit:~$ ls -a
-  .  ..  .bash_logout  .bashrc  .profile
bandit1@bandit:~$ cat ./-
rRGizSaX8Mk1RTb1CNQoXTcYZWU6lgzi
bandit1@bandit:~$ ls -la
total 24
-rw-r-----  1 bandit2 bandit1   33 Oct  5 06:19 -
drwxr-xr-x  2 root    root    4096 Oct  5 06:19 .
drwxr-xr-x 70 root    root    4096 Oct  5 06:20 ..
-rw-r--r--  1 root    root     220 Jan  6  2022 .bash_l
ogout
-rw-r--r--  1 root    root    3771 Jan  6  2022 .bashrc
-rw-r--r--  1 root    root     807 Jan  6  2022 .profil
```

Here is the password .

Bandit Level 2 → Level 3

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

To sign up : ssh bandit2@bandit.labs.overthewire.org -p 2220

Password : rRGizSaX8Mk1RTb1CNQoXTcYZWU6lgzi

```
bandit2@bandit:~$ ls -la
total 24
drwxr-xr-x  2 root    root    4096 Oct  5 06:19 .
drwxr-xr-x 70 root    root    4096 Oct  5 06:20 ..
-rw-r--r--  1 root    root     220 Jan  6  2022 .bash_l
ogout
-rw-r--r--  1 root    root    3771 Jan  6  2022 .bashrc
-rw-r--r--  1 root    root     807 Jan  6  2022 .profil
e
-rw-r-----  1 bandit3 bandit2   33 Oct  5 06:19 spaces
in this filename
bandit2@bandit:~$ cat spaces\ in\ this\ filename
aBZ0W5EmUfAf7kHTQeOwd8bauFJ2lAiG
bandit2@bandit:~$
```

Here is the password .

Bandit Level 3 → Level 4

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

To sign up : ssh bandit3@bandit.labs.overthewire.org -p 2220

Password : aBZ0W5EmUfAf7kHTQeOwd8bauFJ2lAiG

```
bandit3@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  inhere  .profile
bandit3@bandit:~$ cd inhere/
bandit3@bandit:~/inhere$ ls
bandit3@bandit:~/inhere$ ls -a
.  ..  .hidden
bandit3@bandit:~/inhere$ cat <.
./        ../       .hidden
bandit3@bandit:~/inhere$ cat <.hidden
2EW7BBsr6aMMoJ2HjW067dm8EgX26xNe
bandit3@bandit:~/inhere$
```

The password is inside the hidden file .

Bandit Level 4 → Level 5

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

To sign up : ssh bandit4@bandit.labs.overthewire.org -p 2220

Password : 2EW7BBsr6aMMoJ2HjW067dm8EgX26xNe

```
bandit4@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  inhere  .profile
bandit4@bandit:~$ cd inhere/
bandit4@bandit:~/inhere$ ls -a
.  -file00  -file02  -file04  -file06  -file08
..  -file01  -file03  -file05  -file07  -file09
bandit4@bandit:~/inhere$ file ./*
./-file00: data
./-file01: data
./-file02: data
./-file03: data
./-file04: data
./-file05: data
./-file06: data
./-file07: ASCII text
./-file08: data
./-file09: data
bandit4@bandit:~/inhere$ cat -file07
cat: invalid option -- 'f'
Try 'cat --help' for more information.
bandit4@bandit:~/inhere$ cat ./-file07
lrIWWI6bB37kxfiCQZqUdOIYfr6eEeqR
bandit4@bandit:~/inhere$ ls -l ./-file07
-rw-r----- 1 bandit5 bandit4 33 Oct  5 06:19 ./-file07
bandit4@bandit:~/inhere$
```

Let's go to the next stage .

Here we used ls -a to find out hidden files .

The file command in Linux is used to determine the type of a file ,

directoryname /* option : This is used to display all files filetypes in particular directory.

They also used the dot (.)to say "width from here", from where you stand .

Bandit Level 5 → Level 6

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

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

To sign up : ssh bandit5@bandit.labs.overthewire.org -p 2220

Password : lrIWWI6bB37kxfiCQZqUdOIYfr6eEeqR

Here we must use the find command :

```
bandit5@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  inhere  .profile
bandit5@bandit:~$ ls -a inhere/
.            maybehere02  maybehere06  maybehere10  maybehere14  maybehere18
..           maybehere03  maybehere07  maybehere11  maybehere15  maybehere19
maybehere00  maybehere04  maybehere08  maybehere12  maybehere16
maybehere01  maybehere05  maybehere09  maybehere13  maybehere17
bandit5@bandit:~$ find inhere/ -type f  -size 1033c \! -executable
inhere/maybehere07/.file2
bandit5@bandit:~$ cat inhere/maybehere07/.file2
P4L4vucdmLnm8I7Vl7jG1ApGSfjYKqJU
```

Here is the password .

Use -type f to specify the file type .

Use -size 1033c to specify the bytes we want in the file .

We used \! -executable, to specify that the file is not executable .

Bandit Level 6 → Level 7

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

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

To sign up : ssh bandit6@bandit.labs.overthewire.org -p 2220

Password : P4L4vucdmLnm8I7Vl7jG1ApGSfjYKqJU

```
bandit6@bandit:~$ pwd
/home/bandit6
bandit6@bandit:~$ cd /
bandit6@bandit:/$ pwd
/
bandit6@bandit:/$ find . -size 33c -user bandit7 -group bandit6 | grep bandit7
find: ./etc/ssl/private': Permission denied
find: ./etc/polkit-1/localauthority': Permission denied
find: ./etc/sudoers.d': Permission denied
find: ./etc/multipath': Permission denied
find: ./root': Permission denied
find: ./boot/efi': Permission denied
find: ./var/spool/bandit24': Permission denied
find: ./var/spool/cron/crontabs': Permission denied
find: ./var/spool/rsyslog': Permission denied
find: ./var/lib/ubuntu-advantage/apt-esm/var/lib/apt/lists/partial': Permissio
n denied
find: ./var/lib/snapd/cookie': Permission denied
find: ./var/lib/snapd/void': Permission denied
find: ./var/lib/private': Permission denied
find: ./var/lib/chrony': Permission denied
find: ./var/lib/polkit-1': Permission denied
find: ./var/lib/apt/lists/partial': Permission denied
find: ./var/lib/update-notifier/package-data-downloads/partial': Permission de
nied
find: ./var/lib/amazon': Permission denied
find: ./var/log'./var/lib/dpkg/info/bandit7.password
: Permission denied
```

It wasn't the best way to get what I wanted out of it .

I extracted the file with all the same specifications, and used the grep command to remove bandit7 from the output .

So we will use 2>/dev/null , to prevent too much output and output only what we want .

```
bandit6@bandit:/$ find . -size 33c -user bandit7 -group bandit6 2>/dev/null
./var/lib/dpkg/info/bandit7.password
bandit6@bandit:/$ cat ./var/lib/dpkg/info/bandit7.password
z7WtoNQU2XfjmMtWA8u5rN4vzqu4v99S
bandit6@bandit:/$
```

This is better .

Bandit Level 7 → Level 8

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

To sign up : ssh bandit7@bandit.labs.overthewire.org -p 2220

Password : z7WtoNQU2XfjmMtWA8u5rN4vzqu4v99S

```
bandit7@bandit:~$ ls
data.txt
bandit7@bandit:~$ cat data.txt | grep "millionth"
millionth       TESKZC0XvTetK0S9xNwm25STk5iWrBvP
bandit7@bandit:~$
```

The first way .

Here we used grep command, to output the millionth line, to output the password .

The second method is to use vi or nano and then extract the line by searching inside the file :

```
  GNU nano 6.2                    data.txt
hardball        ntMy6waJkKIMaNWArCQyglJxSTsT29L6
Bridgett        lDWjKSjDSq1uhkbasRMyEI1xZkyz7CEZ
umbrage e607e9PYkhslYiGJg2Frrw4lnbkUSFym
afterwords      dmPWQO1BPrpSUhBrND2rEGpzfjscpYYp
juicers RotyYHNYQm6Ve746ni8Aw0Xryb38a8Ec
studios UYfrwVAIFBpfx2ac5xxc3ojmbB2gbUl2
sapling ACh9cpoujPRY03fdQJU69yAY29VC5kYq
Baptist's       lwakVGOuu5Erz3lNXO7khgydeLz1jXt4
debate's        5idFcEMQ0YMiusYRP5VC7CqKrgHrWLjJ
vend    eIYS4d5Y66hsW2YhCgodVtuzBPYkKErS
sermon's        F5B9EHIT1tI56VfczeBtEHWGlSTS0SWl
hoe's   HhTKj3PgrBCYxn8BAHgWg3aYyQPwNs4b
huffy   WJDdimZChuTlmWvX1f00KQgSd3DI63in
millionth       TESKZC0XvTetK0S9xNwm25STk5iWrBvP
Roderick's      zbqXMhsLoFPqc2Mf0TJwI7H6KXp75PSi
cleanup's       ptKYlwR6eOUNk2TM9jxcsg225y6CQZVa
lassie's        Bg8NBxvkuwpoyzw56P3TnFnSEqCFGpDl
holocaust       wABXfALkBM2VhI9RawZOcS2eoOiB7oMy
chews   ROA8sUGIev8xK8lZ0k8rtqBgwTEGNRIN
saddle's        XpyEOvkchTpDXuybqDl3dcigT2n4uJ3l
prohibits       GbtwcIu8dF5OG0essji6mlSM2MgGHtRN
polyglot        VAe4AU5glWwtbT2ENtt80emlmMwh81pJ
sirens  k9tSSGJnkJA3GELIyXytizjPp9kjtaLw
hurling 0mN5hAbJcnZF1mHubgoNxx5vgdIcXxgb
Alan    ynSAA3vOE3ajzzVL1BOmPQnNxISyO7Eh
salaams UPj9uj2sh5XkX7QQcasNVfpNorBxju2h
Evans   pzsxflE0qI9qedZcjhzKTY9rCDNHNPAv
Search [millionth]: millionth
^G Help     M-C Case Sen M-B Backward ^P Older    ^T Go To Line
^C Cancel   M-R Reg.exp. ^R Replace   ^N Newer
```

The second way .

Bandit Level 8 → Level 9

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

To sign up : ssh bandit8@bandit.labs.overthewire.org -p 2220

Password : TESKZC0XvTetK0S9xNwm25STk5iWrBvP

```
bandit8@bandit:~$ ls -a
.  ..  .bash_logout  .bashrc  data.txt  .profile
bandit8@bandit:~$ sort data.txt | uniq -u
EN632PlfYiZbn3PhVK3XOGSlNInNE00t
bandit8@bandit:~$
```

Here is the new password .

The sort command in Unix-based systems is used to sort the contents of a text file called data.txt in alphabetical order. The sort command can also be used to sort in reverse alphabetical order, numerically, and by other criteria.

The uniq -u command in Unix-based systems is used to print only unique lines from a text file. The -u option tells the uniq command to print only lines that have not been seen before.

Bandit Level 9 → Level 10

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

To sign up : ssh bandit9@bandit.labs.overthewire.org -p 2220

Password : EN632PlfYiZbn3PhVK3XOGSlNInNE00t .

```
bandit9@bandit:~$ ls
data.txt
bandit9@bandit:~$ less -2 data.txt
"data.txt" may be a binary file.  See it anyway?

[1]+  Stopped                 less -2 data.txt
bandit9@bandit:~$ cat data.txt | grep ==
grep: (standard input): binary file matches
bandit9@bandit:~$
```

When he looked into the matter further, he learned that the file was not readable at all, so I tried to search using the grep command, but it did not work, so :

Uses of the strings command ; The strings command is a Unix/Linux command-line utility used to extract and display printable strings from non-text files.

```
bandit9@bandit:~$ strings data.txt | grep ====
x]T=========== theG)"
=========== passwordk^
=========== is
=========== G7w8LIi6J3kTb8A7j9LgrywtEUlyyp6s
bandit9@bandit:~$
```

Here is the new password .

Bandit Level 10 → Level 11

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

To sign up : ssh bandit10@bandit.labs.overthewire.org -p 2220

Password : G7w8LIi6J3kTb8A7j9LgrywtEUlyyp6s

```
bandit10@bandit:~$ base64 data.txt
VkdobElIQmhjM04zYjNKa0lHbHpJRFo2VUdWNmFVeGtVakpTUzA1a1RsbEdUbUk
yYmxaRFMzcHdh
R3hZU0VKTkNnPT0K
bandit10@bandit:~$ cat data.txt
VGhlIHBhc3N3b3JkIGlzIDZ6UGV6aUxkUjJSS05kTllGTmI2blZDS3pwaGxYSEJ
NCg==
bandit10@bandit:~$ base64
^C
bandit10@bandit:~$ base64  -h
base64: invalid option -- 'h'
Try 'base64 --help' for more information.
bandit10@bandit:~$ base64  --help
Usage: base64 [OPTION]... [FILE]
Base64 encode or decode FILE, or standard input, to standard ou
tput.

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short opt
ions too.
  -d, --decode          decode data
  -i, --ignore-garbage  when decoding, ignore non-alphabet char
```

Here we should have used — help .

So we have to decode the file using -d :

```
bandit10@bandit:~$ ls
data.txt
bandit10@bandit:~$ base64 -d data.txt
The password is 6zPeziLdR2RKNdNYFNb6nVCKzphlXHBM
bandit10@bandit:~$ echo "Thanks for following_Abdelwahab_Shandy"
Thanks for following_Abdelwahab_Shandy
bandit10@bandit:~$
bandit10@bandit:~$
bandit10@bandit:~$
```