**Useful Unix Commands**

**Prepared for**

**OverthewireBandit**

**Levels**

Prepared by

**Bashetty Arun Kumar**

**Master of Computer Security**

**22-December-2018**

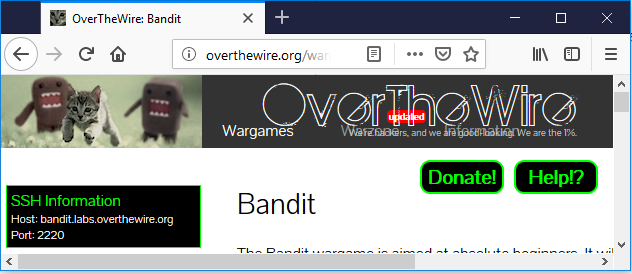
**EXECUTIVE SUMMARY**

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

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

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

**Here is the Proof:**



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

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

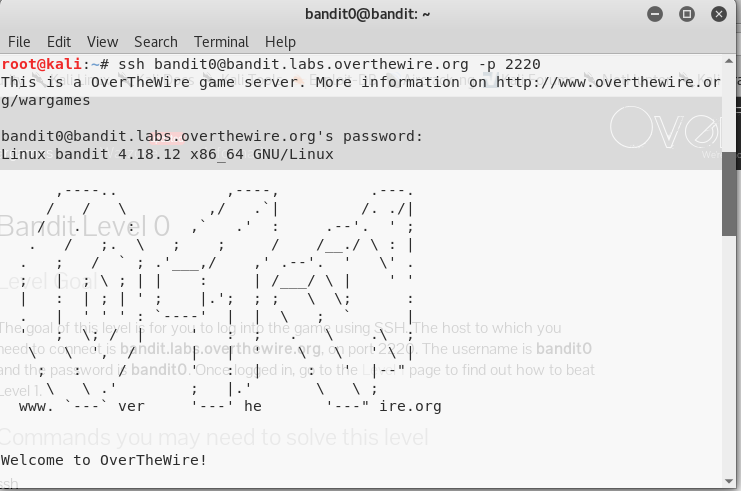
**Bandit Level 0**

﻿The goal of this level is for you to log into the game using SSH. The host to which you need to connect is **bandit.labs.overthewire.org**, on port 2220. The username is **bandit0** and the password is **bandit0**. Once logged in, go to the [Level 1](http://overthewire.org/wargames/bandit/bandit1.html) page to find out how to beat Level 1.

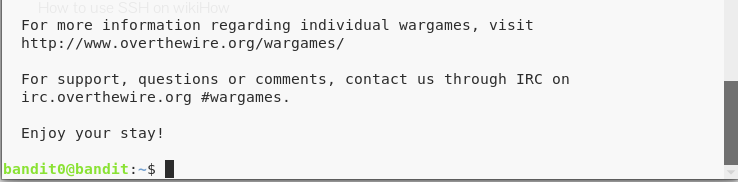
Using ﻿[Secure Shell (SSH) on Wikipedia](http://en.wikipedia.org/wiki/Secure_Shell), I found that how can I connect using ssh command.

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

Password: **bandit0**.



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

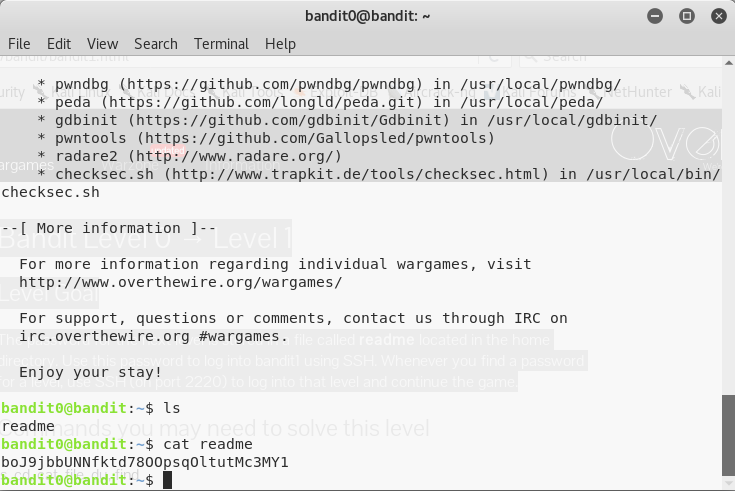


**Bandit Level 0 → Level 1**

## **Level Goal**

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

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



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

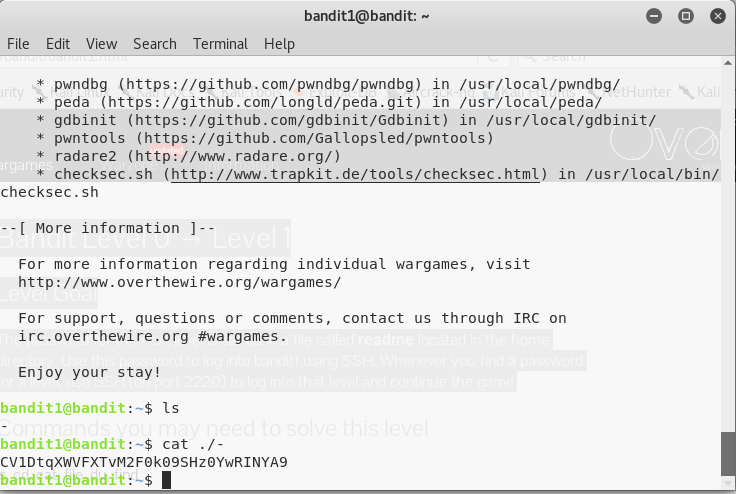
Password for next Level is:

**boJ9jbbUNNfktd78OOpsqOltutMc3MY1**

***Bandit Level 1 → Level 2***

## **Level Goal**

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



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

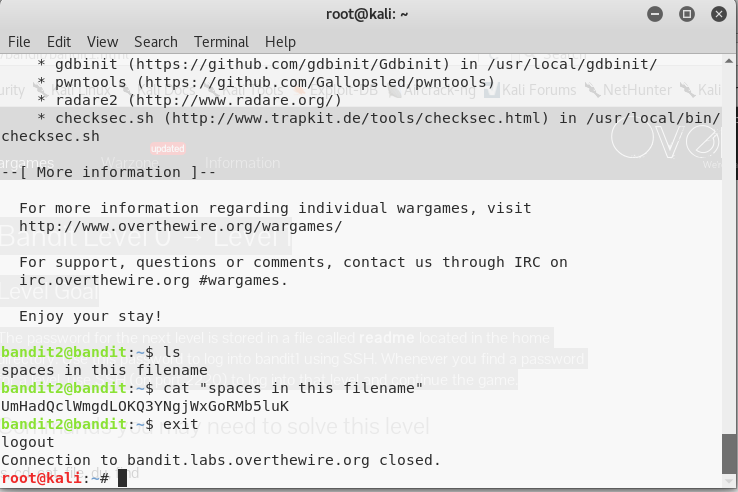
Password for next Level is:

**CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9**

# Bandit Level 2 → Level 3

## **Level Goal**

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



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

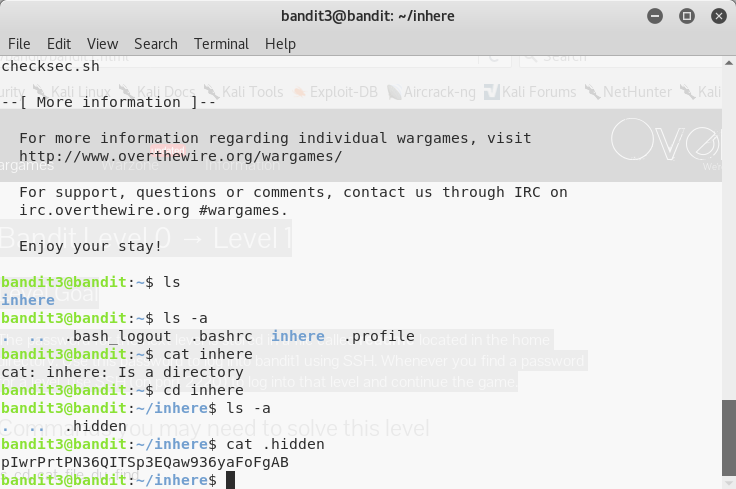
Password for next Level is:

**UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK**

# Bandit Level 3 → Level 4

## **Level Goal**

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



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

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

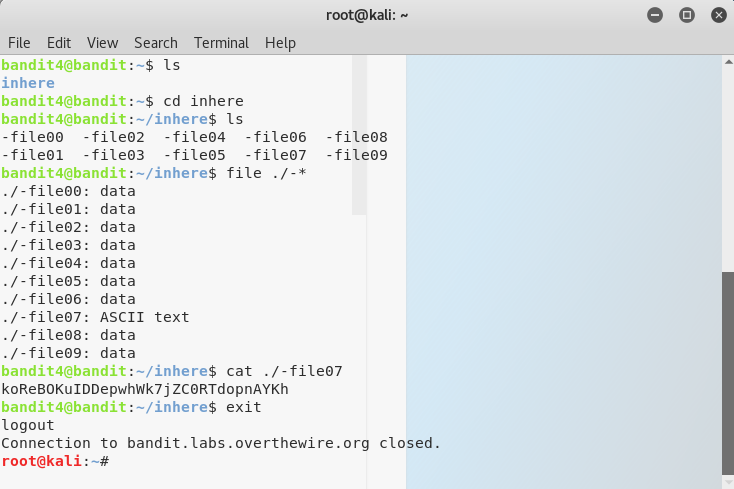
Password for next Level is:

**pIwrPrtPN36QITSp3EQaw936yaFoFgAB**

# Bandit Level 4 → Level 5

## **Level Goal**

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



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

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

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

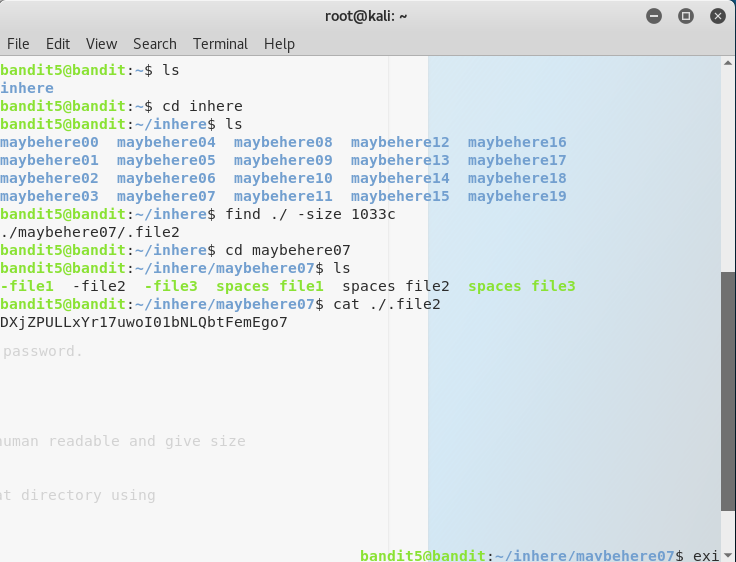
**koReBOKuIDDepwhWk7jZC0RTdopnAYKh**

# Bandit Level 5 → Level 6

## **Level Goal**

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

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



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

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

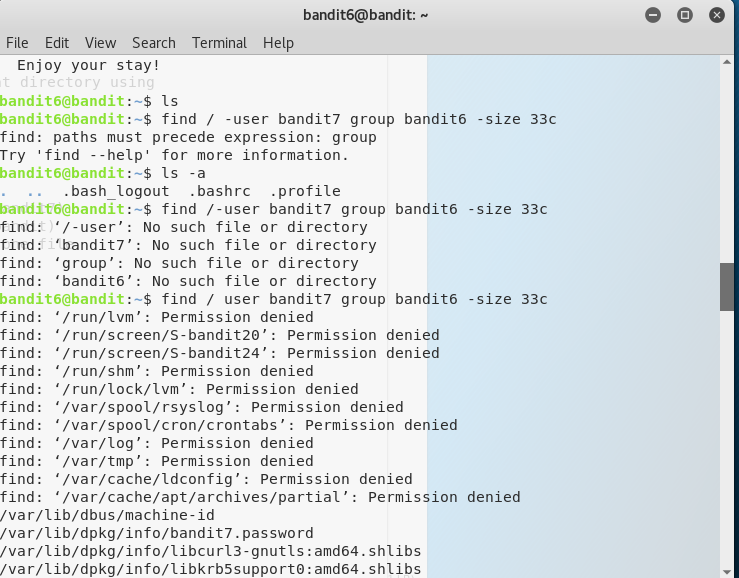
The Password is: **DXjZPULLxYr17uwoI01bNLQbtFemEgo7**

# Bandit Level 6 → Level 7

## **Level Goal**

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

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



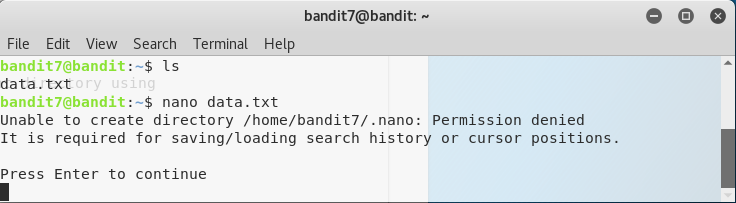


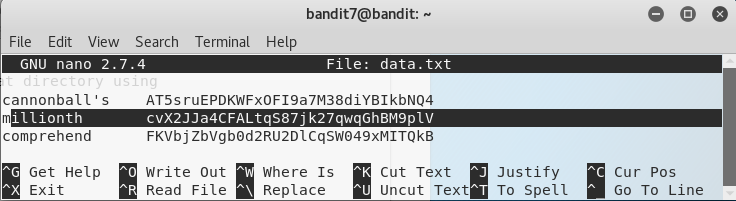
Password for the next Level is: **HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs**

# Bandit Level 7 → Level 8

## **Level Goal**

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





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

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

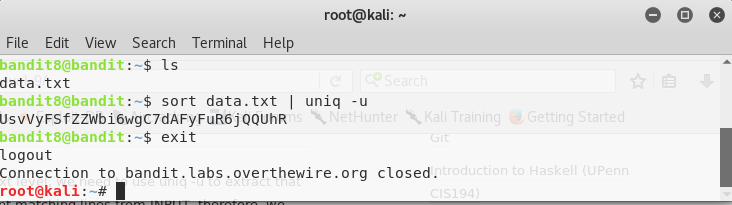
So, I searched the word using CNTRL+R and type millionth,

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

# Bandit Level 8 → Level 9

## **Level Goal**

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



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

# Sort:

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

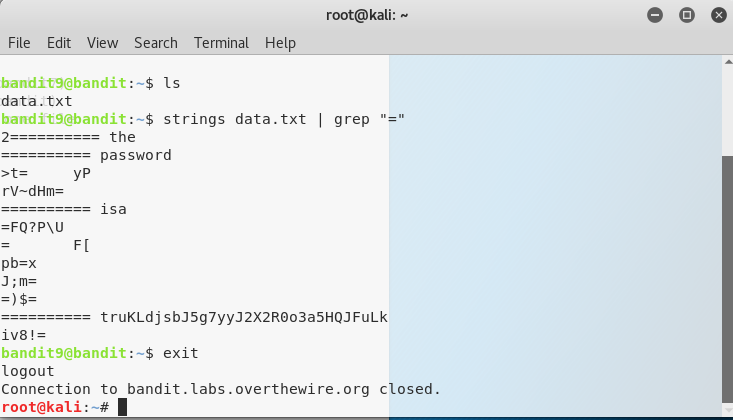
# Password for the next Level is:

# UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR

# Bandit Level 9 → Level 10

## **Level Goal**

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



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

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

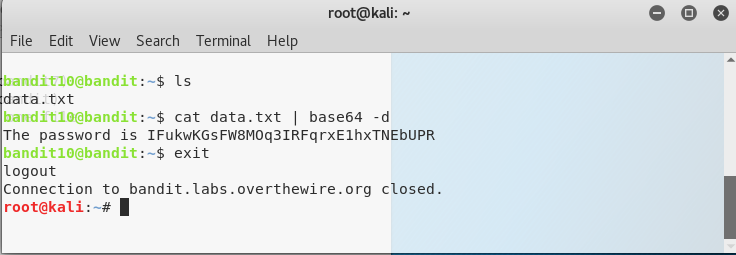
The password for the next Level is:

**truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk**

# Bandit Level 10 → Level 11

## **Level Goal**

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



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

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

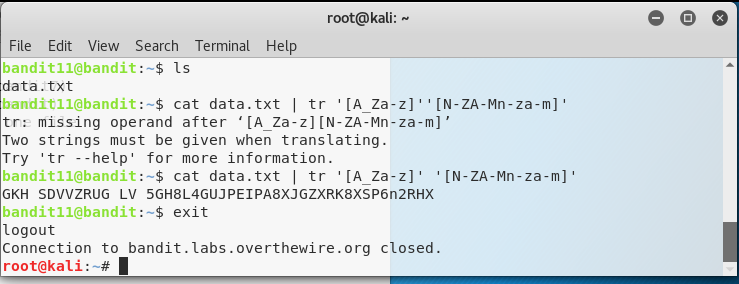
Base64 is a group of similar binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation.

The password the next level: **IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR**

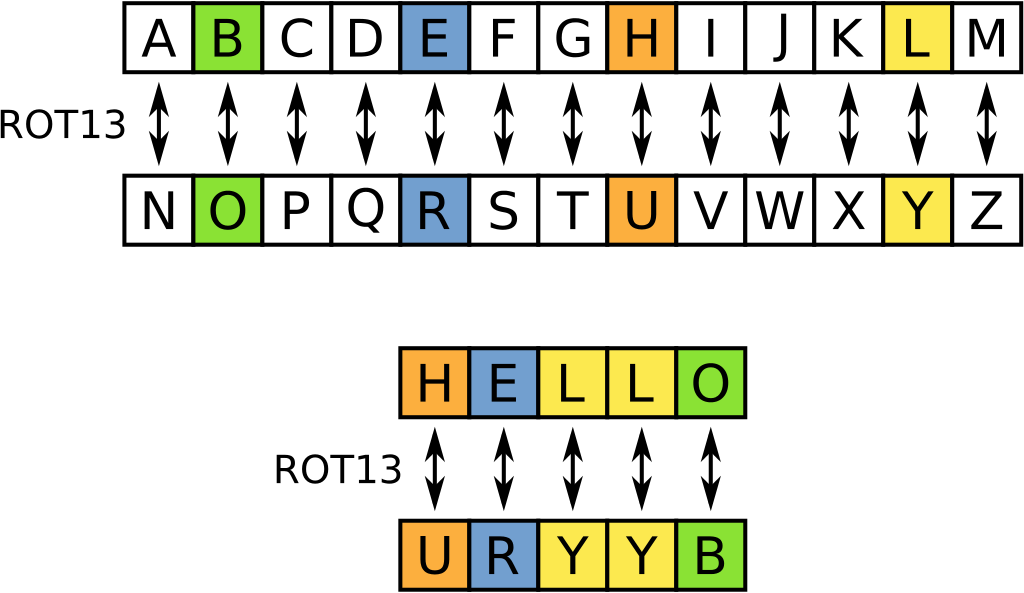
# Bandit Level 11 → Level 12

## **Level Goal**

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



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

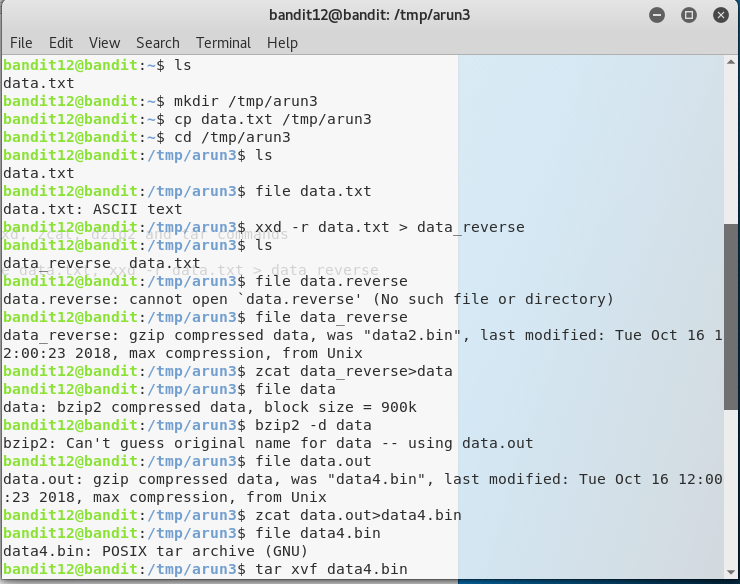


The password for the next Level is: **5Te8Y4drgCRfCx8ugdwuEX8KFC6k2EUu**

# Bandit Level 12 → Level 13

## **Level Goal**

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



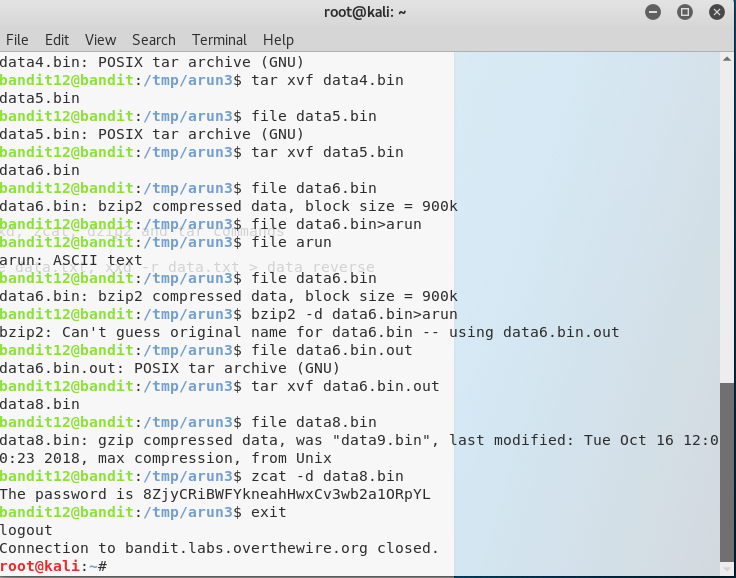
For this Level, we need to know about, Xxd, zcat, bzip2 and tar xvf commands

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

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

Bzip2 -d filename (used to bzip2 compressed data)

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



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

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

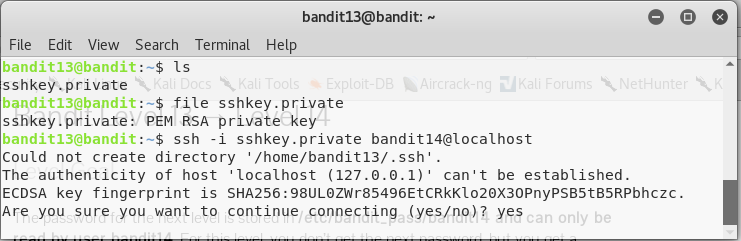
it will finally file showing ASCII password for the next Level is:

**8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL**

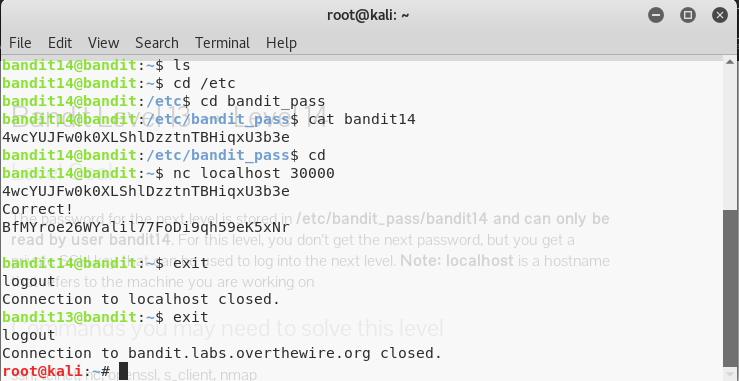
# Bandit Level 13 → Level 14

## **Level Goal**

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



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



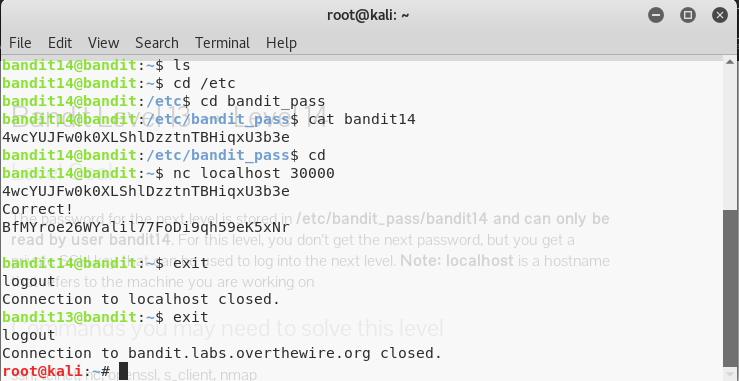
In this above question, it is showing that the password will be inside the /etc/bandit\_pass/bandit14.

The password is**: 4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e**

# Bandit Level 14 → Level 15

## **Level Goal:**

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



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

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

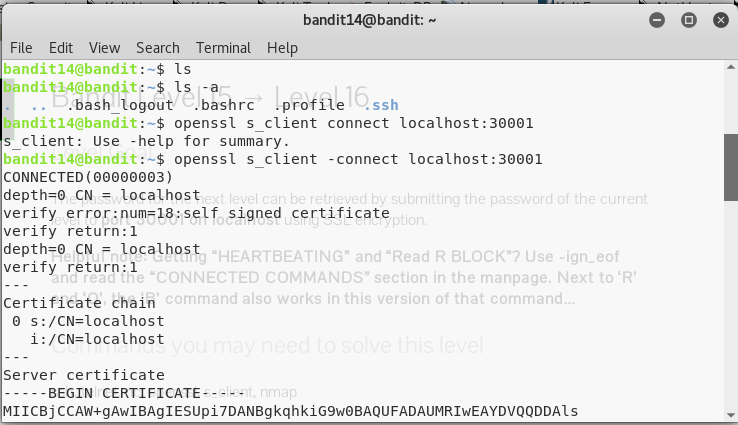
Use the nc command with localhost 30000,

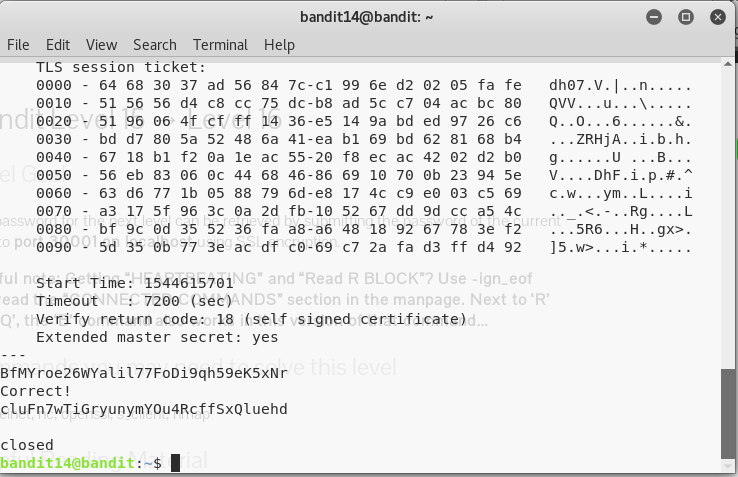
#### The password is: **BfMYroe26WYalil77FoDi9qh59eK5xNr**

# Bandit Level 15 → Level 16

## **Level Goal:**

## The password for the next level can be retrieved by submitting the password of the current level to **port 30001 on localhost** using SSL encryption.





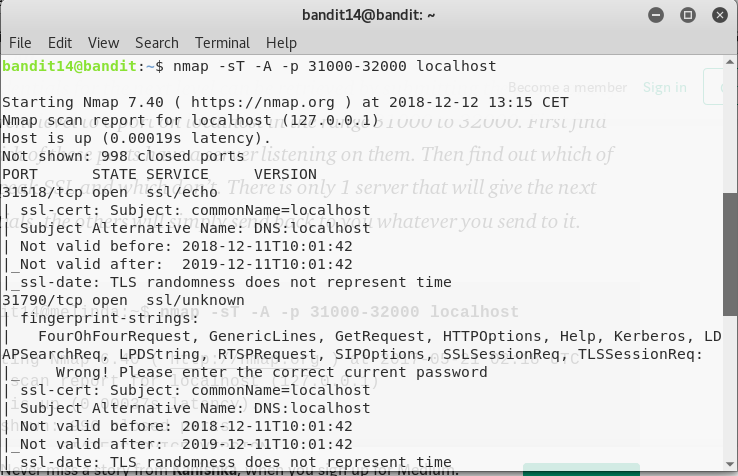
In this level, we can use SSL encryption, OpenSSL is a multi-platform, open source SSL/TLS toolkit, for c\_client **“(-connect host:port)”**

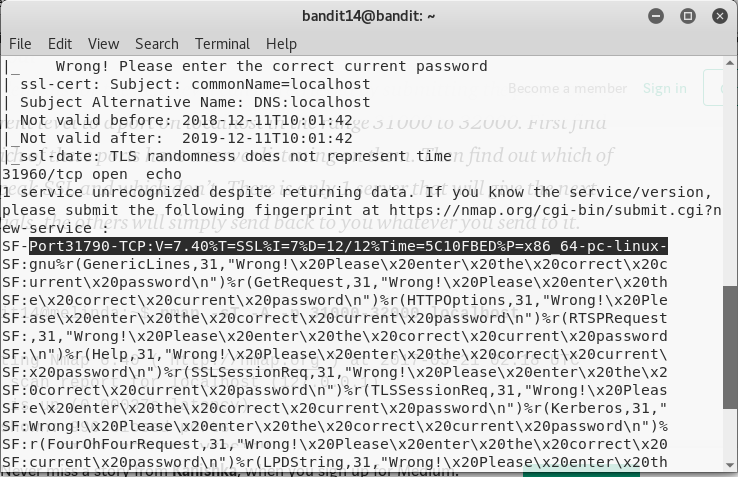
Using s\_connect, localhost as a username with the port number 30001, you can check the password. Password is: **cluFn7wTiGryunymYOu4RcffSxQluehd**

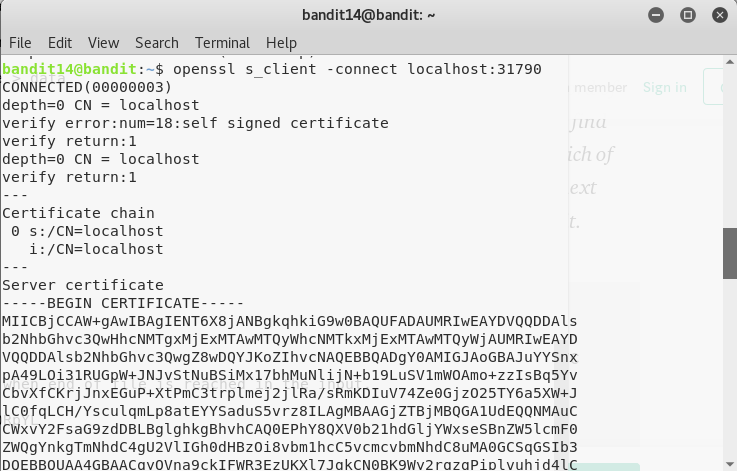
# Bandit Level 16 → Level 17

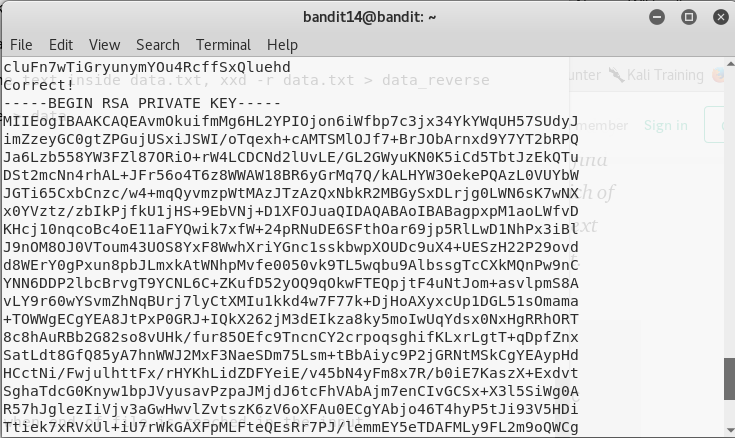
## **Level Goal:**

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









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

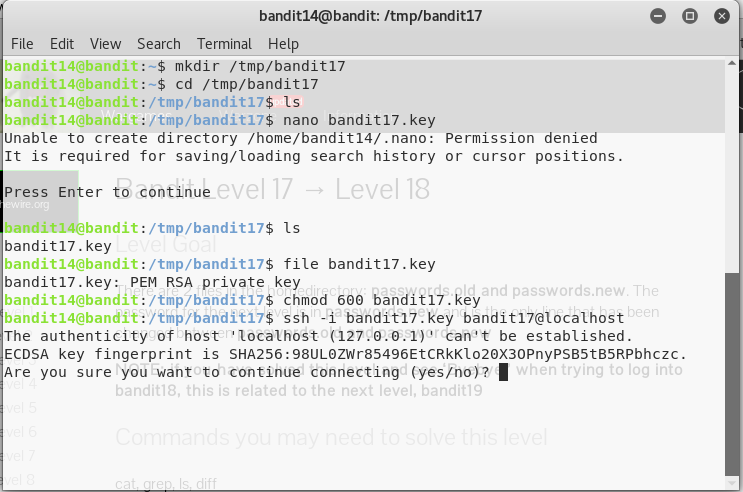
Between host number 31000-32000, only one port number give credentials and you can see the port number:31790 is having credentials,

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

# Bandit Level 17 → Level 18

## **Level Goal**

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





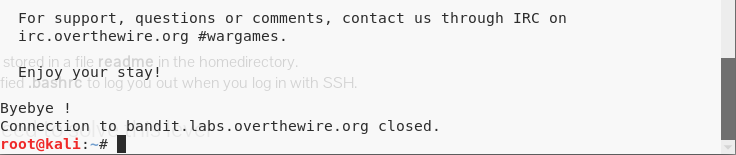
# Here, we can see passwords and use both password for level 18 , it will work only one password, for this level, you must create a directory inside /tmp/bandit17=> bandit17.key

# Next password is : kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd

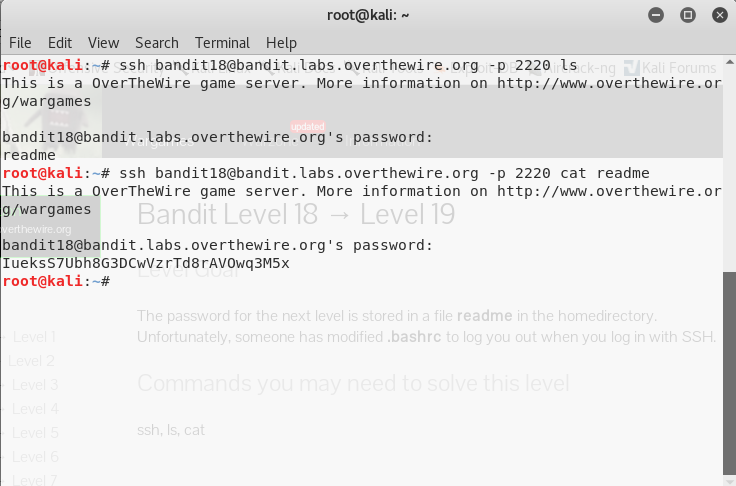
# Bandit Level 18 → Level 19

## **Level Goal**

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



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



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

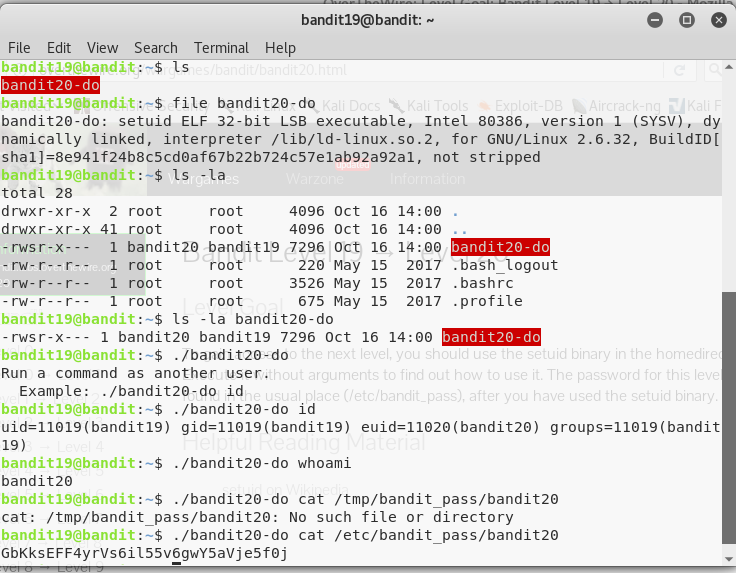
But unfortunately, someone has modified in **.bashrc**, when you ssh for connecting to next level, use **ls** command it will shows readme file and use **cat readme**, it will show Password for next level.

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

# Bandit Level 19 → Level 20

## **Level Goal:**

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



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

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

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

# Bandit Level 20 → Level 21

## **Level Goal:**

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

**Server Side:**

# 

# Client Side:

# 

# On terminal 1 (host), start netcat as server: nc -nvlp 44444.

# On terminal 2 (client/setuid), run suconnect on same port: ./suconnect 44444.

# Back to terminal 1, send the password of current level.

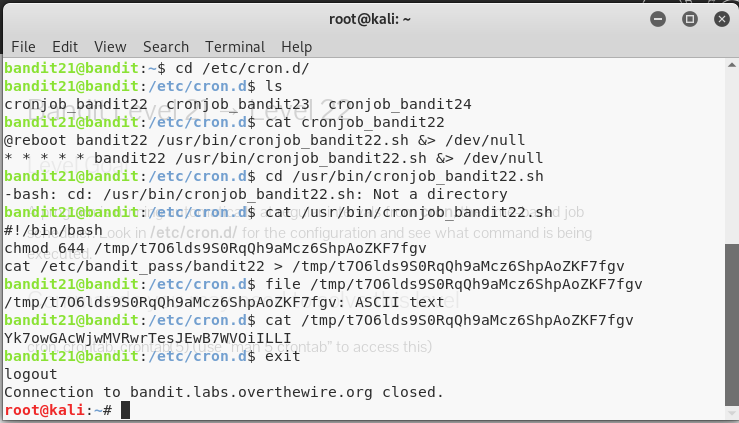
# Suconnect will show you the password.

# Password for the next Level is: gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr

# Bandit Level 21 → Level 22

## **Level Goal**

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



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

It contains some files and inside crinjpb\_bandit22 seem to be might interested and it shows a location if cronjob\_bandit22.sh script.

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

Password for the next level is:

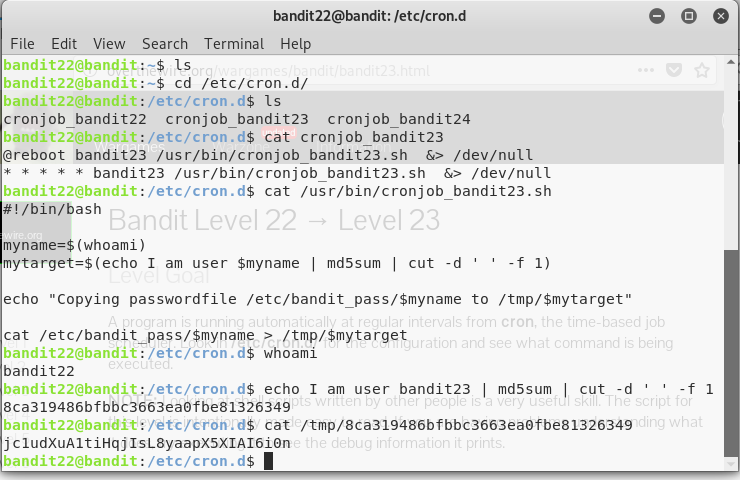
**Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI**

# Bandit Level 22 → Level 23

## **Level Goal:**

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

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



We can open first /etc/cron.d it contains some files and in that cronjob\_bandit23 seems to different because we already checked the bandit22 file,

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

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

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

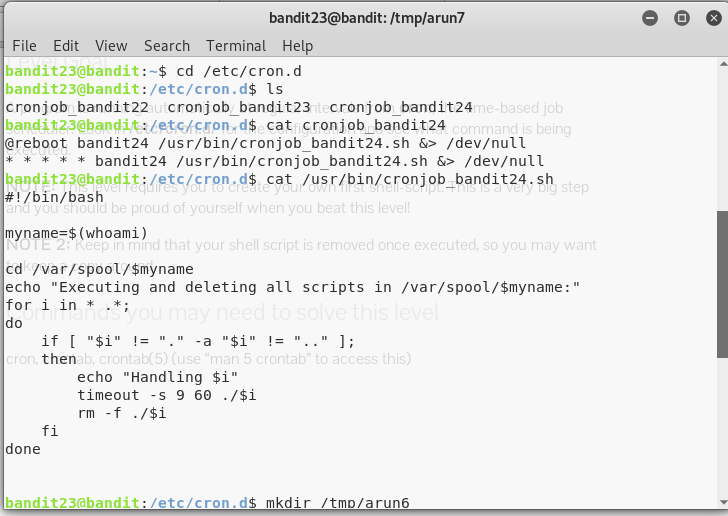
# Bandit Level 23 → Level 24

## **Level Goal**

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

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

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

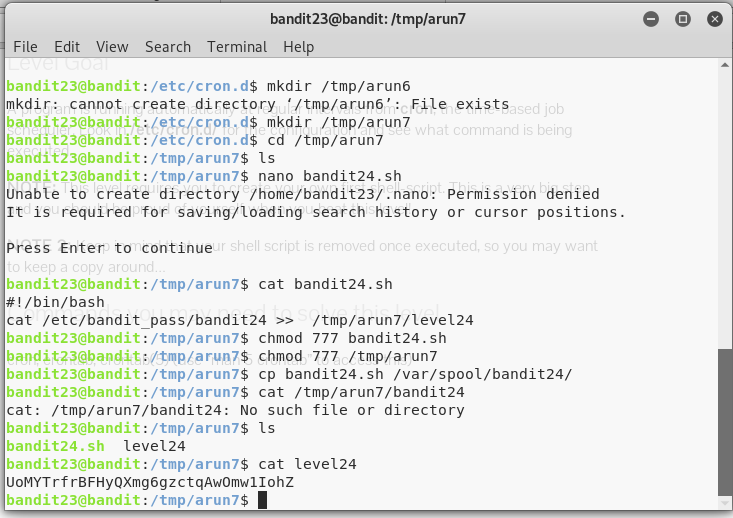


We can open the /etc/cron.d path, we found some files like last 2 levels and in that cronjob\_bandit24 is interesting and you will get a path. Using cat command, you can enter cronjob\_bandit24.sh, we can see the first bash script.

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

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

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



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

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

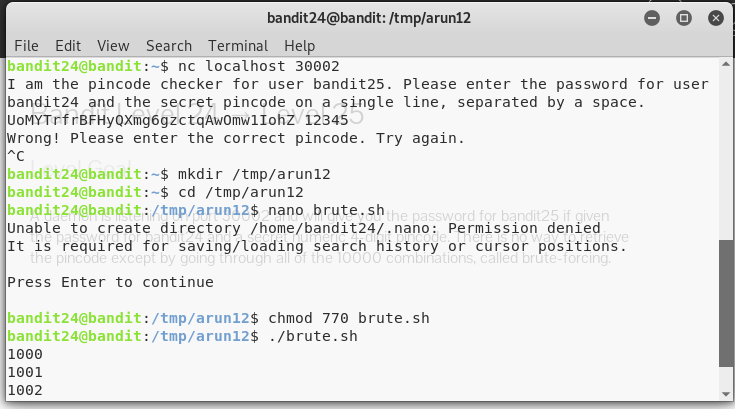
Now you can see inside the /tmp/arun7 , you can see the 2 files, using cat command open the level24 to see the password

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

# Bandit Level 24 → Level 25

## Level Goal

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



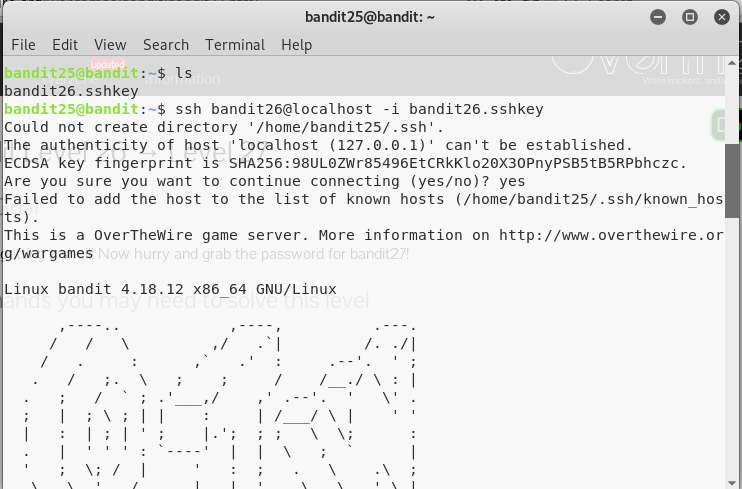
This level need lot of time to get a password, because we are using brute-force algorithm from 0000 to 10000,

For, this step we need a lot of patience and

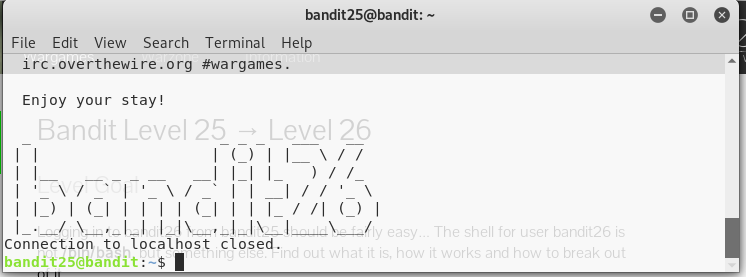
# Bandit Level 25 → Level 26

## **Level Goal**

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



For this role we have to login two times,

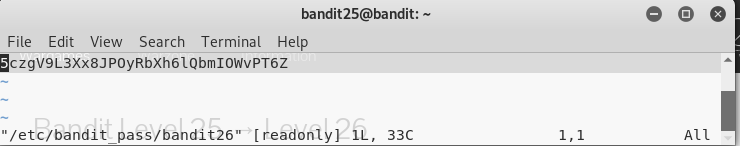


However, after you logged into bandit26, you will be logged out immediately, “Connection to localhost closed.”

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

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

Use command **:e /etc/bandit\_pass/bandit25** and press ENTER,

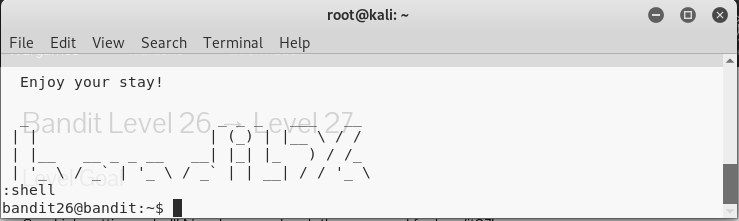


The password for next level is : **5czgV9L3Xx8JPOyRbXh6lQbmIOWvPT6Z**

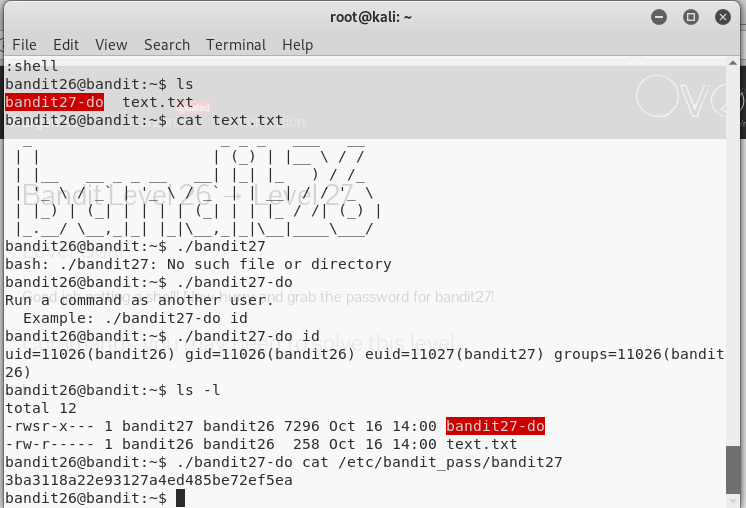
# Bandit Level 26 → Level 27

## **Level Goal**

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



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



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

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

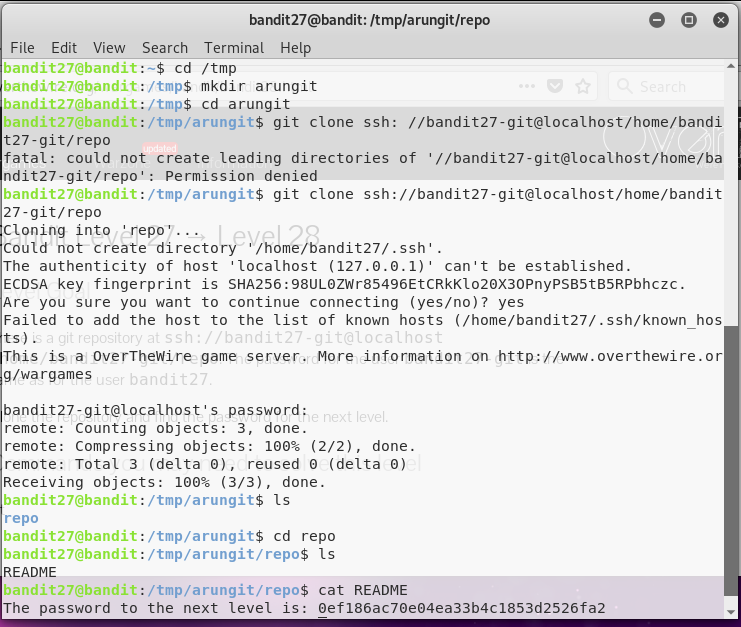
Password for the next level is: **3ba3118a22e93127a4ed485be72ef5ea**

# Bandit Level 27 → Level 28

## **Level Goal**

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

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



Now we need to work with git.

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

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

Use cat command to display the password.

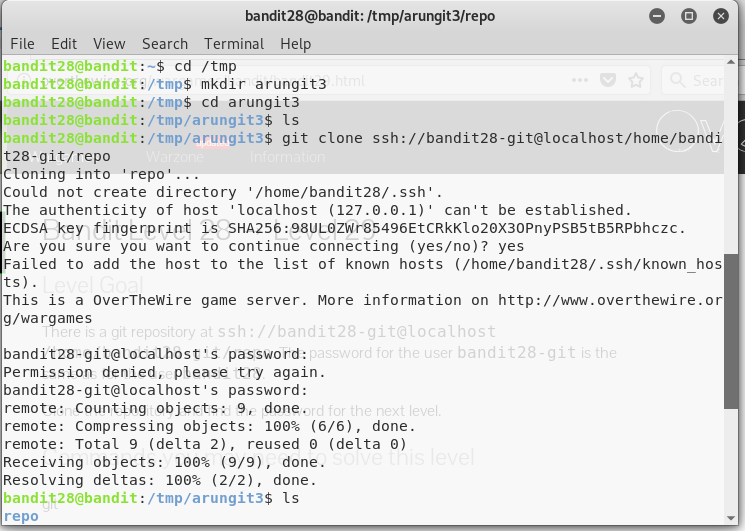
Password for the next level is : **0ef186ac70e04ea33b4c1853d2526fa2**

# Bandit Level 28 → Level 29

## **Level Goal**

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

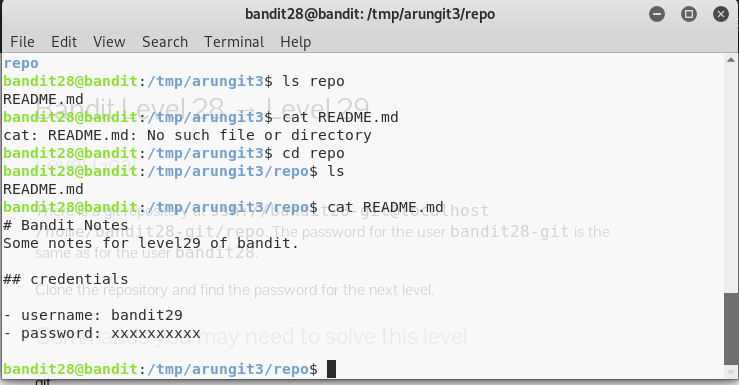
Clone the repository and find the password for the next level.



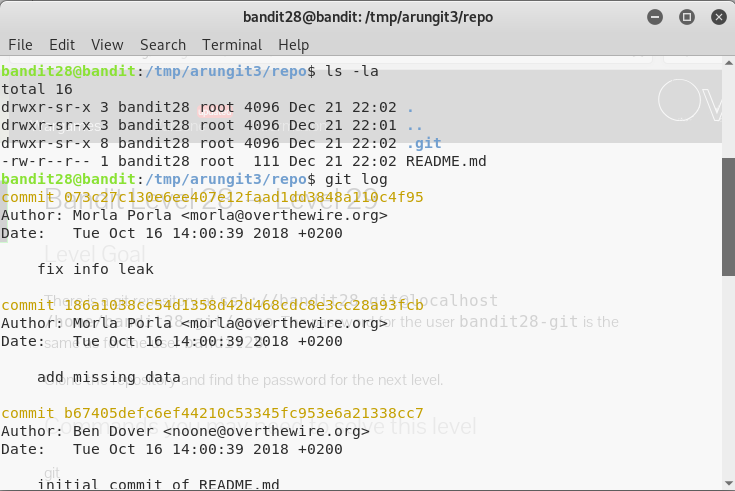
After login in using bandit27 key, change the directory to /tmp and create a directory as I created arungit3 ,

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

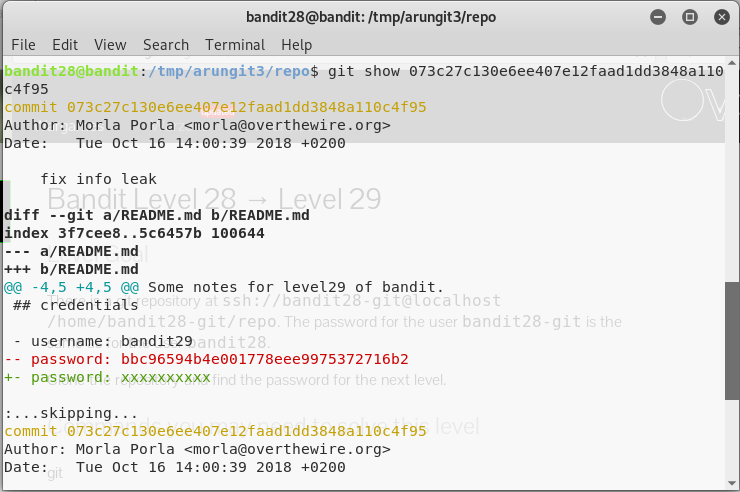
We can check inside the files using ls command and it will display the repo directory.



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



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



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

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

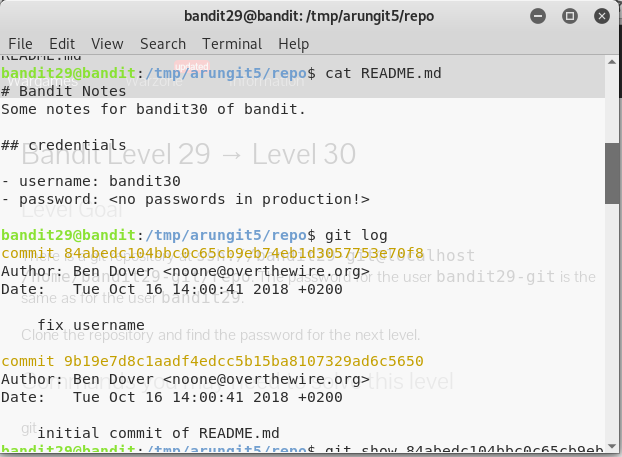
Password for the next level is **: bbc96594b4e001778eee9975372716b2**

# Bandit Level 29 → Level 30

## Level Goal

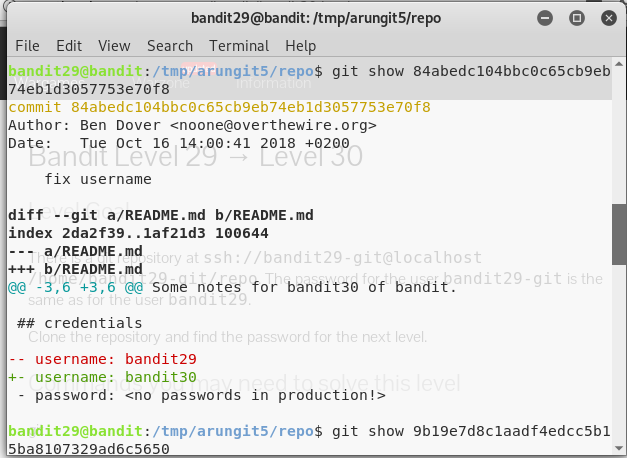
There is a git repository at ssh://bandit29-git@localhost/home/bandit29-git/repo. The password for the user bandit29-git is the same as for the user bandit29.

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

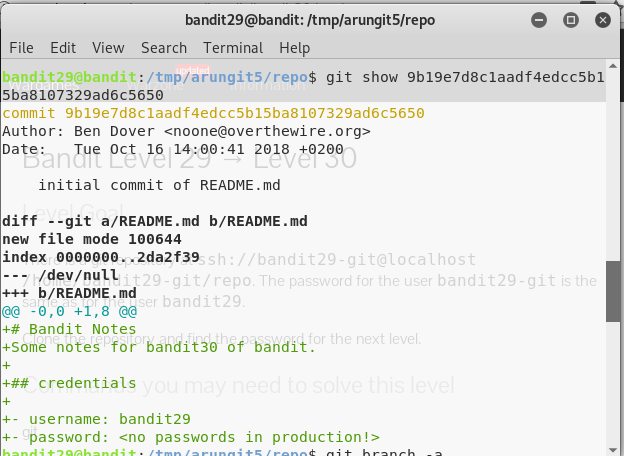


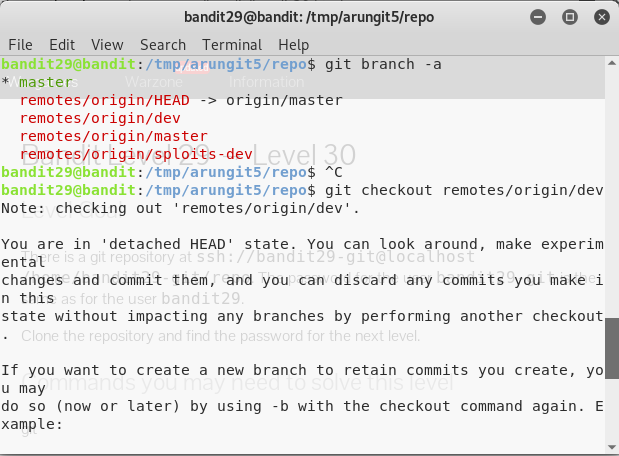
After using git clone command , we can check the README file inside arunfit5/repo file, here it is showing “<no passwords in production!>”

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



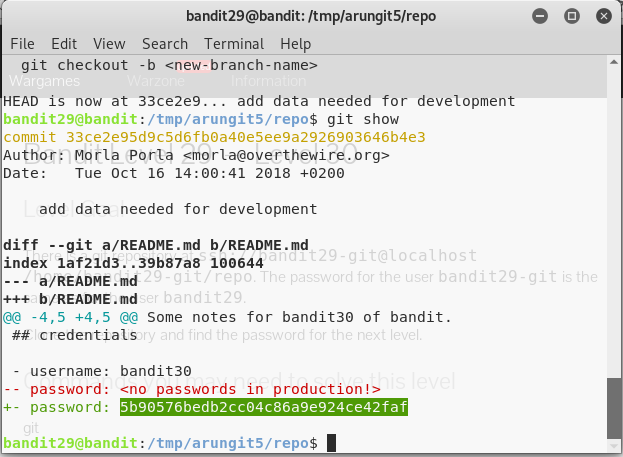
Using git show command with the commit , you can check the first password and it is showing same as first one.





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

You can check the remotes/origin/dev file,



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

Next level password is : **5b90576bedb2cc04c86a9e924ce42faf**

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

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

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