

## Bandit Level 27 → Level 28

SSH Parameters	
Server:	bandit.labs.overthewire.org
Port:	2220

Website URLs	
Level 27→28	<a href="#">OverTheWire: Level Goal: Bandit Level 27 → Level 28</a>
Level 28→29	<a href="#">OverTheWire: Level Goal: Bandit Level 28 → Level 29</a>

Passwords		
Level	User Name	Password
Level 27→28	bandit27	YnQpBuifNMas1hcUFk70ZmqkhUU2EuaS
Level 28→29	bandit28	AVanL161y9rsbcJlsFHuw35rjaOM19nR

The screenshot shows the OverTheWire website interface. The browser address bar displays `https://overthewire.org/wargames/bandit/bandit28.html`. The page title is "Bandit Level 27 → Level 28". The main content area contains the following text:

**Level Goal**

There is a git repository at `ssh://bandit27-git@localhost/home/bandit27-git/repo` via the port 2220. 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.

**Commands you may need to solve this level**

`git`

The sidebar on the left contains a list of levels: Level 0, Level 0 → Level 1, Level 1 → Level 2, Level 2 → Level 3, Level 3 → Level 4, Level 4 → Level 5, Level 5 → Level 6, and Level 6 → Level 7. The "SSH Information" section in the sidebar lists the host as `bandit.labs.overthewire.org` and the port as `2220`. The "Donate!" button is highlighted in green.

```
bandit27@bandit:~$ id && whoami
bandit27@bandit:~$ #Execution of id and whoami commands to determine user ID and bandit level.
bandit27@bandit:~$ #Execution of pwd command to determine current working directory and ls -la to view contents of current working directory. The -l option/switch outputs directory/file metadata and the -a option outputs hidden files/directories. Hidden files/directories are demarked with a period [.] preceding the hidden directory/file.
bandit27@bandit:~$ id && whoami
uid=11027(bandit27) gid=11027(bandit27) groups=11027(bandit27)
bandit27
bandit27@bandit:~$ #Execution of pwd command to determine current working directory and ls -la to view contents of current working directory. The -l option/switch outputs directory/file metadata and the -a option outputs hidden files/directories. Hidden files/directories are demarked with a period [.] preceding the hidden directory/file.
bandit27@bandit:~$ #Execution of pwd command to determine current working directory and ls -la to view contents of current working directory. The -l option/switch outputs directory/file metadata and the -a option outputs hidden files/directories. Hidden files/directories are demarked with a period [.] preceding the hidden directory/file.
bandit27@bandit:~$ pwd && ls -la
/home/bandit27
total 20
drwxr-xr-x 2 root root 4096 Apr 23 18:04 .
drwxr-xr-x 70 root root 4096 Apr 23 18:05 ..
-rw-r--r-- 1 root root 220 Jan 6 2022 .bash_logout
-rw-r--r-- 1 root root 3771 Jan 6 2022 .bashrc
-rw-r--r-- 1 root root 807 Jan 6 2022 .profile
bandit27@bandit:~$
```

```
bandit27@bandit: /tmp/AC_bandit27$ #The git clone command will deposit the git repository to the directory we specify on our local computer. This machine enables writing to the /tmp directory. As such we utilize the mkdir command to create a subdirectory, titled AC_bandit27, under /tmp. Next we navigate to /tmp/AC_bandit27. Next, we execute the pwd command to determine if the current working directory is /tmp/AC_bandit27. These commands are sequentially executed via the && operator.
bandit27@bandit: /tmp/AC_bandit27$ #The git clone command will deposit the git repository to the directory we specify on our local computer. This machine enables writing to the /tmp directory. As such we utilize the mkdir command to create a subdirectory, titled AC_bandit27, under /tmp. Next we navigate to /tmp/AC_bandit27. Next, we execute the pwd command to determine if the current working directory is /tmp/AC_bandit27. These commands are sequentially executed via the && operator.
bandit27@bandit: /tmp/AC_bandit27$ #The git clone command will deposit the git repository to the directory we specify on our local computer. This machine enables writing to the /tmp directory. As such we utilize the mkdir command to create a subdirectory, titled AC_bandit27, under /tmp. Next we navigate to /tmp/AC_bandit27. Next, we execute the pwd command to determine if the current working directory is /tmp/AC_bandit27. These commands are sequentially executed via the && operator.
bandit27@bandit: /tmp/AC_bandit27$ mkdir -v /tmp/AC_bandit27 && cd /tmp/AC_bandit27 && pwd
mkdir: created directory '/tmp/AC_bandit27'
/tmp/AC_bandit27
bandit27@bandit: /tmp/AC_bandit27$
```

```
#####
#Per Over the Wire Bandit Level 27 ----> Level 28 directions, harvesting the key requires utilizing the Global Information Tracker [git] suite of commands to clone the repository at ssh://bandit27-git@localhost/home/bandit27-git/repo [on port 2220] and harvest the password.

#Git [different from GitHub website] is an open source software tool utilized to manage and combine the [software] code authored by a team of individuals. In example, an application such as VLC Media Player is large, rich in functionality, and composed of multiple units of code all joined together in a single application. The git application, through a suite of commands, provides a code repository and the ability of personnel [i.e. Development Operations [DevOps]] to incorporate their contributions to the master code base. Additionally, git provides ancillary functionality such as the ability to roll back code to previous versions.

#Solving this level requires utilization of the git clone command. This command copies the prescribed code repository to our local computer. The code repository is contained in a directory. We then utilize standard bash commands, such as cd, ls, and cat, to navigate and access the contents of the code repository.
#####
```



```

bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$ #To view contents of the /tmp/AC_bandit27/repo directory we leverage the ls command and
bandit27@bandit:/tmp/AC_bandit27/repo$ #nd invoke the -l option/switch to output file/directory metadata and the -a option/switch to output hidden directories [deno
bandit27@bandit:/tmp/AC_bandit27/repo$ #ted by a period [.] at the first position of the file/directory titles]
bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$ ls -la
total 16
drwxrwxr-x 3 bandit27 bandit27 4096 Aug 22 20:58 .
drwxrwxr-x 3 bandit27 bandit27 4096 Aug 22 20:58 ..
drwxrwxr-x 8 bandit27 bandit27 4096 Aug 22 20:58 .git
-rw-rw-r-- 1 bandit27 bandit27 68 Aug 22 20:58 README
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$ #We utilize the file command to determine the file, and data type, of the README file
bandit27@bandit:/tmp/AC_bandit27/repo$ #contianed in /tmp/AC_bandit27/repo
bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$ file README
README: ASCII text
bandit27@bandit:/tmp/AC_bandit27/repo$

```

```

bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$ #Based on the output of the file command README is an ASCII text command. To investi
bandit27@bandit:/tmp/AC_bandit27/repo$ #gate if this file contains the password we utilize the cat command to view its contents.
bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$ cat README
The password to the next level is: AVanL161y9rsbcJIsFHuw35rjaOM19nR

```

```

bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$ #Based on the output of the cat command the README file contains the bandit28 passwor
bandit27@bandit:/tmp/AC_bandit27/repo$ #d. To extract this file we cat the README file and pipe the output to the cut command and invoke the -d option/switch to del
bandit27@bandit:/tmp/AC_bandit27/repo$ #ineate on space " ", and the -f option/switch to extract the 8th field [bandit28 password].
bandit27@bandit:/tmp/AC_bandit27/repo$ #####
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$
bandit27@bandit:/tmp/AC_bandit27/repo$ cat README | cut -d " " -f 8
AVanL161y9rsbcJIsFHuw35rjaOM19nR

```

\*\*\*\*\*

## Level 28 —> Level 29 Password

\*\*\*\*\*

AVanL161y9rsbcJIsFHuw35rjaOM19nR