Bandit – Stage by Stage explanation

Level 0

All you have to do here is ssh onto the bandit server. This can be done with the command `$ ssh -p 2220 bandit0@bandit.labs.overthewire.org` and then entering the password `bandit0`

Level 1

The password for the next level is stored in a file on the desktop named 'readme'.

This can be revealed with the `ls` command, and read with `cat readme`.

The password for bandit1 is `boJ9jbbUNNfktd78OOpsqOltutMc3MY1`

Level 2

The password here is stored in a file called '-'

This can be revealed with the `ls` command, but running `cat -` confuses the system, as `-` normally denotes a flag. We can get around this by using the complete file path, './-'. This gives us the command `cat ./-`

The password for bandit2 is `CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9`

Level 3

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

This cannot be accessed by simply running `cat spaces in this file name`, as the system will register this as four separate files. Instead, we can either use the escape character, '', in front of the spaces, or specify the file name in quotation marks. This gives us either `cat spaces\ in\ this\ filename` or `cat "spaces in this file name"`

The password for bandit3 is `UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK`

Level 4

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

We can change into this directory using `cd inhere`, and then reveal the hidden file, which is called '.hidden'. We can then read the contents of this file with the command `cat .hidden`.

The password for bandit4 is `pIwrPrtPN36QITSp3EQaw936yaFoFgAB`

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.

Again, we can change to the inhere directory using `cd inhere`. Running `ls` then reveals that there are ten files in this directory. As we know that the password is contained in a human readable file, we need to somehow filter the files by filetype. This can be done with the command `find . -type f | xargs file`. This works by finding all the files in the current directory, and passing these as an argument into the file command, which then lists them by type. Doing so reveals to us that 9 of the 10 files are of type 'data', but one is of type 'ASCII text'. Therefore, the password must be in this file.

This works given the relatively small number of files in this directory, but what if it were larger? We could extend this command by piping the output into `grep text`, giving us a complete command of

```
find . -type f | xargs file | grep text
```

This would return only the file marked as ASCII text, omitting all others.

The password for bandit5 is koReBOKuIDDepwhWk7jZC0RTdopnAYKh

Level 6

In this level, we are given three properties of the file containing the password. It is human readable, 1033 bytes and not executable.

Therefore, we need to somehow string these conditions together into a search command. We can again start with `find . -type f`, which is going to recursively find all files of in this directory and subdirectories. However, this gives an unusably long output. At this point, one might think to apply `| xargs file | grep text`, as in the previous level.

This was tried, but still presented the issue of giving an unworkably large number of files. For that reason, I changed the command used, and introduced the `-exec` flag to my initial find statement. This allows us to also run ls with the appropriate flags (to list more information on the file, including file size and whether it is executable). The flags I used were -la; l is the alias of long listing, giving more information, and a is the alias for all, meaning even hidden files are included. From here, I just needed to add a filter so that only files 1033 bytes long were

shown - this is done using the `grep 1033` command. By piping these together, we are left with

```
find . type -f -exec ls -la {} \; | grep 1033
```

The password for bandit6 is DXjZPULLxYr17uwoI01bNLQbtFemEgo7

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

Immediately, we know that we wish to search the entire file system. This can be done with `find /`. As before, we know that we want to look for all files, so we can introduce the flag `-type f`. We can then make use of the `-user`, `-group` and `-size` flags for the find command, giving us:

```
$ find / -user bandit7 -group bandit6 -type f -size 33c
```

This works, but we get a slew of permission denied errors. This makes it difficult to identify where the correct file is. We want to somehow suppress these - this can be done by introducing `2>/dev/null` to the command, giving us a final command of

```
$ find / -user bandit7 -group bandit6 -type f -size 33c 2>/dev/null
```

The bandit7 password is `HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs`