OverTheWire: Bandit Solutions

INTRODUCTION:

OverTheWire's **Bandit** wargame, an excellent resource for learning Linux commands and security concepts.

This guide provides solutions for **Bandit levels 1-20**, explaining the steps needed to progress through each level.

Level 0 → 1: Connecting via SSH

The first level is all about establishing an SSH connection.

- **Username:** `bandit0`
- **Host:** `bandit.labs.overthewire.org`
- **Port:** `2220`

Command to Connect:

ssh bandit0@bandit.labs.overthewire.org -p 2220

Command to Read Password:

```
cat readme
```

BANDIT Level 1 → 2: Reading a File Named -

The password for the next level is stored in a file named -, which can be tricky to read due to its special character.

Command to Read Password:

```
cat ./-
```

BANDIT Level 2 → 3: Handling Spaces in Filenames

The password is inside a file named **"spaces in this filename”.**

Ways to Read the File:

Using escape characters (`\`):

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

Using quotes:

```
cat "spaces in this filename"
```

BANDIT Level 3 → 4: Finding a Hidden File

The password is stored in a **hidden file** within the `inhere` directory.

Commands to Find and Read the Hidden File:

```
cd inhere

ls -a

cat .hidden
```

BANDIT Level 4 → 5: Identifying a Human-Readable File

The password is stored in one of the files in the `inhere` directory. We need to identify which file is **human-readable**.

Steps to Find the File:

Navigate to the directory:

```
cd inhere
```

1. Identify the readable file:
   ```
   file ./*
   ```
2. This will output file types, allowing us to find the one labeled **ASCII text**.

3. Read the correct file:
   ```
   cat ./-file07
   ```

BANDIT Level 5 → 6: Finding a File with Specific Attributes

The next password is stored in a file somewhere under `inhere` with these characteristics:

- **Human-readable**

- **1033 bytes in size**

- **Not executable**

Find and Read the File:

```
find ./inhere -type f -size 1033c ! -executable
```

```
cat ./inhere/maybehere07/.file2
```

BANDIT Level 6 → 7: Searching System-Wide for a File:

Command to Locate the File:

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

BANDIT Level 7 → 8: Extracting a Password from a File

The password is stored in data.txt next to the word **"millionth"**.

Command to Find the Password:

```
grep "millionth" data.txt
```

BANDIT Level 8 → 9: Finding a Unique Line

The password is inside data.txt and appears **only once** in the file.

Command to Extract the Unique Line:

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

BANDIT Level 9 → 10: Extracting Readable Strings

The password is hidden within data.txt among non-readable characters. It is one of the **human-readable strings** that begins with =.

Command to Filter Readable Strings:

```
strings data.txt | grep "="
```

BANDIT Level 10 → 11: Decoding Base64

The password is stored in data.txt as a **Base64-encoded string**. We need to decode it.

Command to Decode:

```
cat data.txt | base64 --decode
```

Bandit Level 11 → 12 👍:

To retrieve the password, we need to decode the text using ROT13. You can use an online tool, but there's also a way to do it directly in Linux using the `tr` command.

Steps:

Display the contents of the file:

```
cat data.txt
```

1. Output:

```
Gur cnffjbeq vf 5Gr8L4qetPEsPk8htqjhRK8XSP6x2RHh
```

2.Apply ROT13 decoding using `tr`:

```
cat data.txt | tr 'A-Za-z' 'N-ZA-Mn-za-m'
```

Bandit Level 12 → 13:

By examining the file, we notice it has been transformed into a hexdump. We need to reverse this process and then determine the type of data it contains to extract the password.

Steps:

1.Convert the hexdump back to its original format:

```
xxd -r data.txt > decoded_file
```

2.Check the file type:

`file decoded_file`.

3.Rename and extract the file according to its format:

```
mv decoded_file decoded.gz
```

```
gunzip decoded.gz
```

PASSWORD:8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL

Bandit Level 13 → 14:

View the SSH private key:

```
cat sshkey.private
```

Use the key to log in as `bandit14`:

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

Once logged in, retrieve the password:

```
cat /etc/bandit_pass/bandit14
```

Bandit Level 14 → 15

Solution:

We can accomplish this using `netcat` (`nc`).

Steps:

1. Send the password to port 30000 using `echo` and `nc`:

   ```
   echo "4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e" | nc localhost 30000
   ```

Bandit Level 16 → 17

Solution:

Scan for open ports in the given range:

```
nmap -sV -p 31000-32000 localhost
```

This will list the services running on the open ports. Look for one that uses SSL or provides a key.

Connect to the identified port using OpenSSL:

```
openssl s_client -connect localhost:31790
```

After entering the previous level's password, the terminal will output an RSA private key.

Retrieve the password for the next level:

```
cat /etc/bandit_pass/bandit17
```

Bandit Level 17 → 18

Solution:

Identify differences between the old and new password files:

```
diff passwords.old passwords.new
```

The output will display the new password.

Bandit Level 18 → 19

Solution:

Execute a non-interactive command to read the password:

```
ssh -p 2220 bandit18@bandit.labs.overthewire.org "cat readme"
```

Bandit Level 19 → 20

Use a program that allows running commands as `bandit20` to access its password.

Solution:

1. **Run the special program provided:**
   ```
   ./bandit20-do cat /etc/bandit_pass/bandit20
   ```