Bandit_11→15

Writeup for Bandit Level 11 → Level 12

Title: Bandit Level 11 - Decoding Rot13 Encoded Data

Introduction

Bandit Level 11 requires users to decode a Rot13 encoded string stored in a file named `data.txt`. The password for the next level is contained within this decoded data. This writeup documents the steps taken to complete Level 11 and retrieve the password for 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 (Rot13).

Methodology

1. **Connect to the Server Using SSH**:
   - Open a terminal and use the `ssh` command to connect to the server.
   - The command used is:

     `ssh bandit11@bandit.labs.overthewire.org -p 2220` .
   - When prompted, enter the password retrieved from Level 10: `dtRI73fZKbORRsDFSGsg2RMnpMVj3qRr`.

```
┌──(pinkman㉿pinkman)-[~]
└─$ ssh bandit11@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


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

bandit11@bandit.labs.overthewire.org's password:
```

2. **Access the Server**:
   - After successfully logging in, you will be in the home directory of the `bandit11` user.
3. **Locate the `data.txt` File**:
   - List the contents of the home directory using the `ls` command.
   - You will see a file named `data.txt`.
4. **Retrieve the Password for Level 12**:
   - Use the `tr` command to decode the Rot13 encoded data in the `data.txt` file: `cat data.txt | tr '[A-Za-z]' '[N-ZA-Mn-za-m]'`
   - The `tr` command translates characters from one set to another. In this case, it shifts each letter by 13 positions, effectively decoding the Rot13 encoding.
   - The decoded output will contain the password for Level 12.

```
bandit11@bandit:~$ ls
data.txt
bandit11@bandit:~$ cat data.txt
Gur cnffjbeq vf 7k16JArUVv5LxVuJfsSVdbbtaHGlw9D4
bandit11@bandit:~$ cat data.txt | tr '[A-Za-z]' '[N-ZA-Mn-za-m]'
The password is 7x16WNeHIi5YkIhWsfFIqoognUTyj9Q4
bandit11@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```

Findings/Results

- The password for Level 12 is: `7×16WNeHIi5YkIhWsfFIqoognUTyj9Q4`

Discussion/Analysis

- Level 11 introduces the challenge of decoding Rot13 encoded data. The `tr` command is essential for performing character translation in Linux.
- This level emphasizes the importance of understanding simple encoding schemes and using the appropriate tools to decode data in a Linux environment.

Conclusion

- Successfully logged into the Bandit game server as `bandit11`.
- Retrieved the password for Level 12 by decoding the Rot13 encoded data in the `data.txt` file using the `tr` command.
- This level reinforces the importance of using text processing commands to decode information from encoded data.

Commands Used

- `ssh bandit11@bandit.labs.overthewire.org -p 2220`: Connect to the server via SSH.
- `ls`: List files in the current directory.
- `cat data.txt | tr '[A-Za-z]' '[N-ZA-Mn-za-m]'`: Decode the Rot13 encoded data in `data.txt`

Screenshots

1. **SSH Connection**:

```
┌──(pinkman㉿pinkman)-[~]
└─$ ssh bandit11@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


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

bandit11@bandit.labs.overthewire.org's password:
```

2. **Retrieving the Password**:

```
bandit11@bandit:~$ ls
data.txt
bandit11@bandit:~$ cat data.txt
Gur cnffjbeq vf 7k16JArUVv5LxVuJfsSVdbbtaHGlw9D4
bandit11@bandit:~$ cat data.txt | tr '[A-Za-z]' '[N-ZA-Mn-za-m]'
The password is 7x16WNeHIi5YkIhWsfFIqoognUTyj9Q4
bandit11@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```

Writeup for Bandit Level 12 → Level 13

Title: Bandit Level 12 - Decoding a Repeatedly Compressed Hexdump

---

Introduction

Bandit Level 12 requires users to decode a hexdump stored in a file named `data.txt`. The hexdump represents a file that has been repeatedly compressed using different methods. The password for the next level is contained within this file after decompression. This writeup documents the steps taken to complete Level 12 and retrieve the password for 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.

Methodology

1. **Connect to the Server Using SSH**:
   - Open a terminal and use the `ssh` command to connect to the server.
   - The command used is:

     `ssh` `bandit12@bandit.labs.overthewire.org` `-p 2220`

   - When prompted, enter the password retrieved from Level 11: `7×16MMeHIl5YkIhWsfFIqoognUryj904`.

2. **Access the Server**:
   - After successfully logging in, you will be in the home directory of the `bandit12` user.
3. **Create a Temporary Directory**:
   - Create a temporary directory under `/tmp` to work in:

     `mkdir /tmp/try`
   - Copy the `data.txt` file to this directory:

     `cp data.txt /tmp/try`

```
bandit12@bandit:~$ ls
data.txt
bandit12@bandit:~$ mkdir /tmp/try && cp data.txt /tmp/try
bandit12@bandit:~$ cd /tmp/try
bandit12@bandit:/tmp/try$ ls
data.txt
```

4. **Convert the Hexdump Back to Binary**:
   - Use the `xxd` command to convert the hexdump back to its original binary form:

     `xxd -r data.txt > data`
5. **Identify and Decompress the File**:
   - Use the `file` command to identify the type of the file:

     `file data`

```
data.txt
bandit12@bandit:/tmp/try$ xxd -r data.txt > data
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: gzip compressed data, was "data2.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 574
```

   - Depending on the output, use the appropriate decompression tool:
     - For `gzip` compressed files:

       `mv data data.gz`

       `gzip -d data.gz`
     - For `bzip2` compressed files:

       `mv data data.bz`

       `bzip2 -d data.bz`

- For `tar` archives:

  `mv data data.tar`

  `tar xf data.tar`

```
bandit12@bandit:~$ ls
data.txt
bandit12@bandit:~$ mkdir /tmp/try && cp data.txt /tmp/try
bandit12@bandit:~$ cd /tmp/try
bandit12@bandit:/tmp/try$ ls
data.txt
bandit12@bandit:/tmp/try$ xxd -r data.txt > data
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: gzip compressed data, was "data2.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 574
bandit12@bandit:/tmp/try$ mv data data.gz
bandit12@bandit:/tmp/try$ ls
data.gz  data.txt
bandit12@bandit:/tmp/try$ gzip -d data.gz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/try$ mv data data.bz
bandit12@bandit:/tmp/try$ ls
data.bz  data.txt
bandit12@bandit:/tmp/try$ bzip2 -d data.bz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: gzip compressed data, was "data4.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 20480
bandit12@bandit:/tmp/try$ mv data data.gz
bandit12@bandit:/tmp/try$ ls
data.gz  data.txt
bandit12@bandit:/tmp/try$ gzip -d data.gz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: POSIX tar archive (GNU)
bandit12@bandit:/tmp/try$ mv data data.tar
bandit12@bandit:/tmp/try$ ls
data.tar  data.txt
bandit12@bandit:/tmp/try$ tar xf data.tar
bandit12@bandit:/tmp/try$ ls
data5.bin  data.tar  data.txt
```

- Repeat the process of identifying and decompressing the file until you reach a file that contains the password.

6. **Retrieve the Password for Level 13**:

   - After several decompression steps, you will eventually find a file containing the password:

     `cat data`

   - The password for Level 13 will be displayed.

```
bandit12@bandit:/tmp/try$ ls
data  data6.tar  data.tar  data.txt
bandit12@bandit:/tmp/try$ file data
data: ASCII text
bandit12@bandit:/tmp/try$ cat data
The password is FO5dwFsc0cbaIiH0h8J2eUks2vdTDwAn
bandit12@bandit:/tmp/try$ exit
logout
Connection to bandit.labs.overthewire.org closed.
```

Findings/Results

- The password for Level 13 is: `FO5dwFsc0cbaIiH0h8J2eUks2vdTDwAn`

Discussion/Analysis

- Level 12 introduces the challenge of working with a hexdump of a file that has been repeatedly compressed using different methods. The `xxd`, `file`, and various decompression commands are essential for this task.
- This level emphasizes the importance of understanding file types and using the appropriate tools to decompress and extract data in a Linux environment.

Conclusion

- Successfully logged into the Bandit game server as `bandit12`.
- Retrieved the password for Level 13 by converting the hexdump back to binary and repeatedly decompressing the file using the appropriate tools.
- This level reinforces the importance of using a combination of commands to handle complex file manipulations.

Commands Used

- `ssh bandit12@bandit.labs.overthewire.org -p 2220`: Connect to the server via SSH.
- `mkdir /tmp/try`: Create a temporary directory.
- `cp data.txt /tmp/try`: Copy the `data.txt` file to the temporary directory.
- `xxd -r data.txt > data`: Convert the hexdump back to binary.
- `file data`: Identify the type of the file.
- `gzip -d data.gz`, `bzip2 -d data.bz`, `tar xf data.tar`: Decompress the file based on its type.
- `cat data`: Display the contents of the final file containing the password.

Screenshots

1. **SSH Connection**:

```
┌──(pinkman㉿pinkman)-[~]
└─$ ssh bandit12@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


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

bandit12@bandit.labs.overthewire.org's password:
```

2. **Decompression Process**:

```
bandit12@bandit:~$ ls
data.txt
bandit12@bandit:~$ mkdir /tmp/try && cp data.txt /tmp/try
bandit12@bandit:~$ cd /tmp/try
bandit12@bandit:/tmp/try$ ls
data.txt
bandit12@bandit:/tmp/try$ xxd -r data.txt > data
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: gzip compressed data, was "data2.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 574
bandit12@bandit:/tmp/try$ mv data data.gz
bandit12@bandit:/tmp/try$ ls
data.gz  data.txt
bandit12@bandit:/tmp/try$ gzip -d data.gz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/try$ mv data data.bz
bandit12@bandit:/tmp/try$ ls
data.bz  data.txt
bandit12@bandit:/tmp/try$ bzip2 -d data.bz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: gzip compressed data, was "data4.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 20480
bandit12@bandit:/tmp/try$ mv data data.gz
bandit12@bandit:/tmp/try$ ls
data.gz  data.txt
bandit12@bandit:/tmp/try$ gzip -d data.gz
bandit12@bandit:/tmp/try$ ls
data  data.txt
bandit12@bandit:/tmp/try$ file data
data: POSIX tar archive (GNU)
bandit12@bandit:/tmp/try$ mv data data.tar
bandit12@bandit:/tmp/try$ ls
data.tar  data.txt
bandit12@bandit:/tmp/try$ tar xf data.tar
bandit12@bandit:/tmp/try$ ls
data5.bin  data.tar  data.txt
```

```
bandit12@bandit:/tmp/try$ file data5.bin
data5.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/try$ mv data5.bin data.tar
bandit12@bandit:/tmp/try$ ls
data.tar  data.txt
bandit12@bandit:/tmp/try$ tar xf data.tar
bandit12@bandit:/tmp/try$ ;s
-bash: syntax error near unexpected token `;'
bandit12@bandit:/tmp/try$ ls
data6.bin  data.tar  data.txt
bandit12@bandit:/tmp/try$ file data6.bin
data6.bin: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/try$ mv data6.bin data6.bz
bandit12@bandit:/tmp/try$ ls
data6.bz  data.tar  data.txt
bandit12@bandit:/tmp/try$ bzip2 -d data6.bz
bandit12@bandit:/tmp/try$ ls
data6  data.tar  data.txt
```

3. **Final Decompression and Password Retrieval**:

```
bandit12@bandit:/tmp/try$ file data6
data6: POSIX tar archive (GNU)
bandit12@bandit:/tmp/try$ mv data6 data6.tar
bandit12@bandit:/tmp/try$ ls
data6.tar  data.tar  data.txt
bandit12@bandit:/tmp/try$ tar xf data6.tar
bandit12@bandit:/tmp/try$ ls
data6.tar  data8.bin  data.tar  data.txt
bandit12@bandit:/tmp/try$ file data8.bin
data8.bin: gzip compressed data, was "data9.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 49
bandit12@bandit:/tmp/try$ file data8.bin
data8.bin: gzip compressed data, was "data9.bin", last modified: Thu Sep 19 07:08:15 2024, max compression, from Unix, original size modulo 2^32 49
bandit12@bandit:/tmp/try$ mv data8.bin data.gz
bandit12@bandit:/tmp/try$ ls
data6.tar  data.gz  data.tar  data.txt
bandit12@bandit:/tmp/try$ gzip -d data.gz
bandit12@bandit:/tmp/try$ ls
data  data6.tar  data.tar  data.txt
bandit12@bandit:/tmp/try$ file data
data: ASCII text
bandit12@bandit:/tmp/try$ cat data
The password is FO5dwFsc0cbaIiH0h8J2eUks2vdTDwAn
bandit12@bandit:/tmp/try$ exit
logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```

Writeup for Bandit Level 13 → Level 14

Title: Bandit Level 13 - Using a Private SSH Key to Access the Next Level

---

Introduction

Bandit Level 13 provides a private SSH key that can be used to log into the next level as `bandit14`. The password for Level 14 is stored in a file that can only be read by `bandit14`. This writeup documents the steps taken to complete Level 13 and retrieve the password for Level 14.

Level Goal

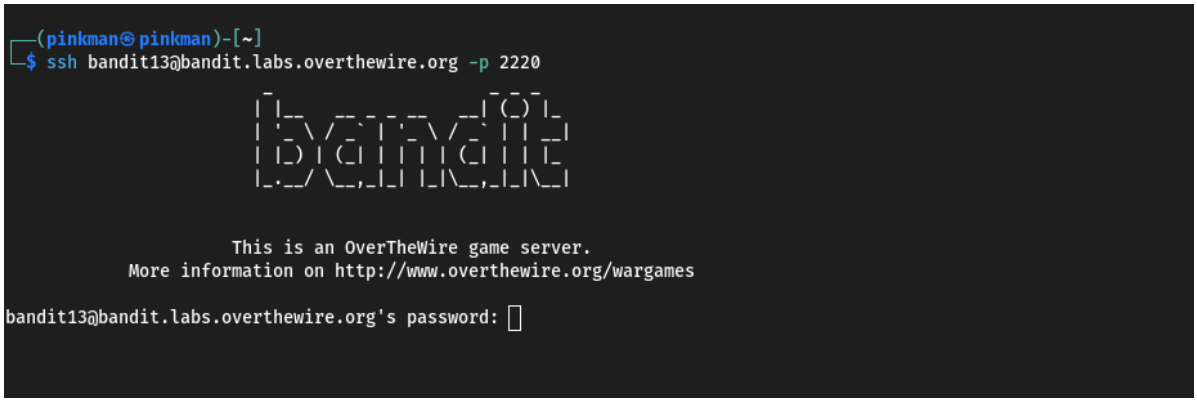
- The password for the next level is stored in `/etc/bandit_pass/bandit14` and can only be read by user `bandit14`.
- You are provided with a private SSH key to log into the next level.

Methodology

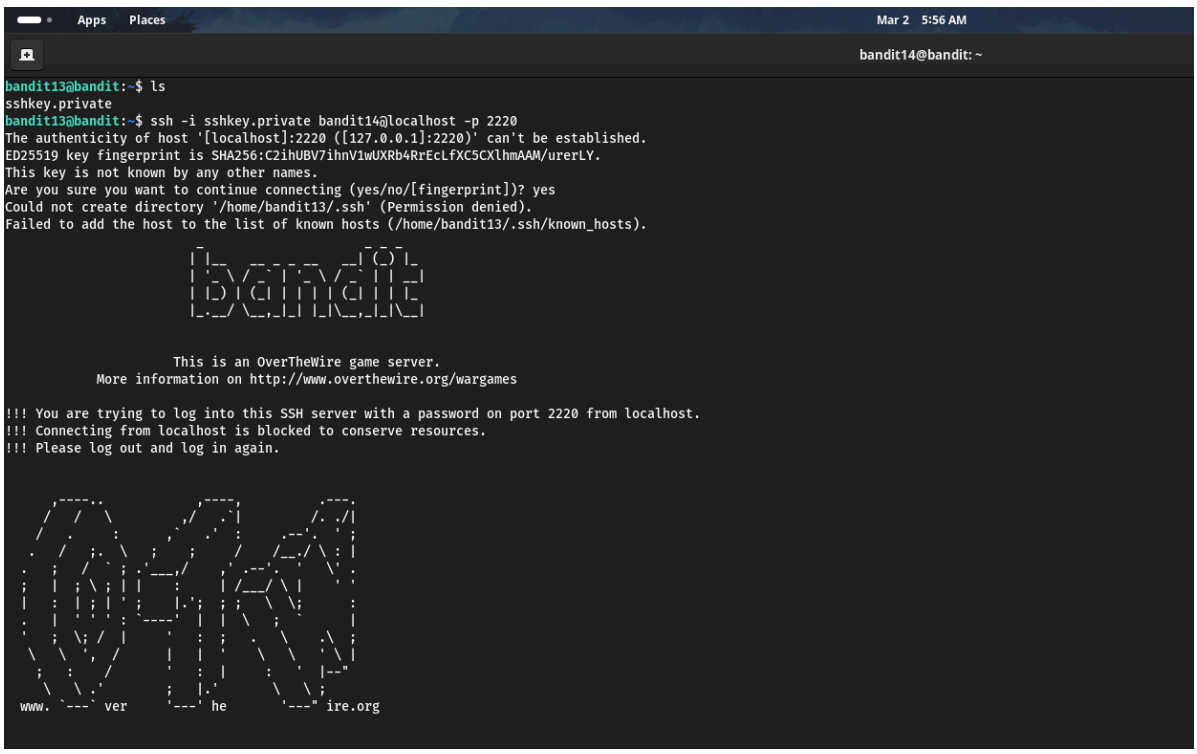
1. **Connect to the Server Using SSH**:
   - Open a terminal and use the `ssh` command to connect to the server.
   - The command used is:

     `ssh bandit13@bandit.labs.overthewire.org -p 2220`

   - When prompted, enter the password retrieved from Level 12: `F05dmfsc0cbai1H0H82JcUk52vGTDwAn`.



2. **Access the Server**:
   - After successfully logging in, you will be in the home directory of the `bandit13` user.
3. **Locate the Private SSH Key**:
   - List the contents of the home directory using the `ls` command.
   - You will see a file named `sshkey.private`.
4. **Use the Private SSH Key to Log into `bandit14`**:
   - Use the `ssh` command with the private key to log into `bandit14`:
     `ssh -i sshkey.private bandit14@localhost -p 2220`
   - When prompted to confirm the authenticity of the host, type `yes`.



5. **Retrieve the Password for Level 14**:
   - Once logged in as `bandit14`, read the password from the file `/etc/bandit_pass/bandit14`:
     `cat /etc/bandit_pass/bandit14`

- The password for Level 14 will be displayed.

```
bandit14@bandit:~$ cat  /etc/bandit_pass/bandit14
MU4VWeTyJk8ROof1qqmcBPaLh7lDCPvS
bandit14@bandit:~$
```

Findings/Results

- The password for Level 14 is: `NU4VWeryJk8Roof1qqmcBPaLh7lDCPvS`

Discussion/Analysis

- Level 13 introduces the use of private SSH keys for authentication. The private key allows you to log into the next level without needing a password.
- This level emphasizes the importance of understanding SSH key-based authentication and how to use private keys to access restricted resources.

Conclusion

- Successfully logged into the Bandit game server as `bandit13`.
- Used the provided private SSH key to log into `bandit14`.
- Retrieved the password for Level 14 by reading the file `/etc/bandit_pass/bandit14`.
- This level reinforces the importance of SSH key-based authentication and accessing restricted files.

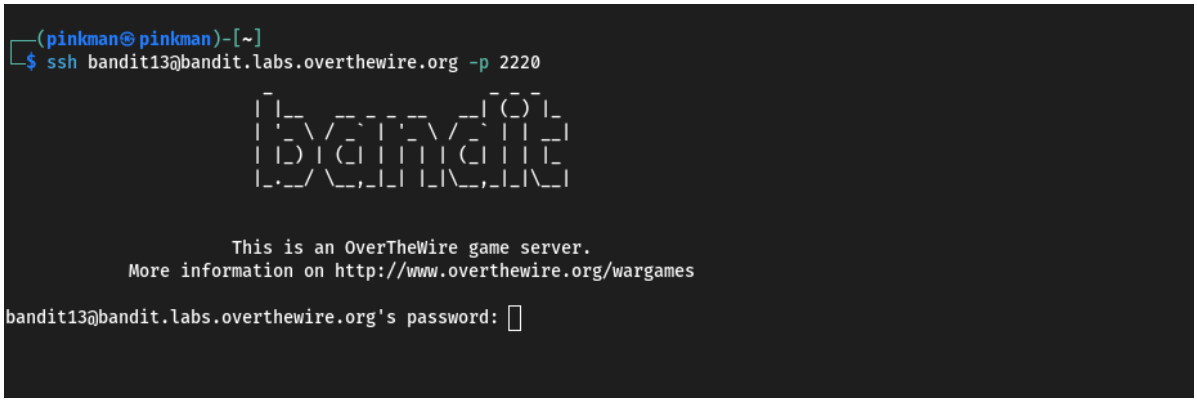
Commands Used

- `ssh bandit13@bandit.labs.overthewire.org -p 2220`: Connect to the server via SSH.
- `ls`: List files in the current directory.
- `ssh -i sshkey.private bandit14@localhost -p 2220`: Log into `bandit14` using the private SSH key.

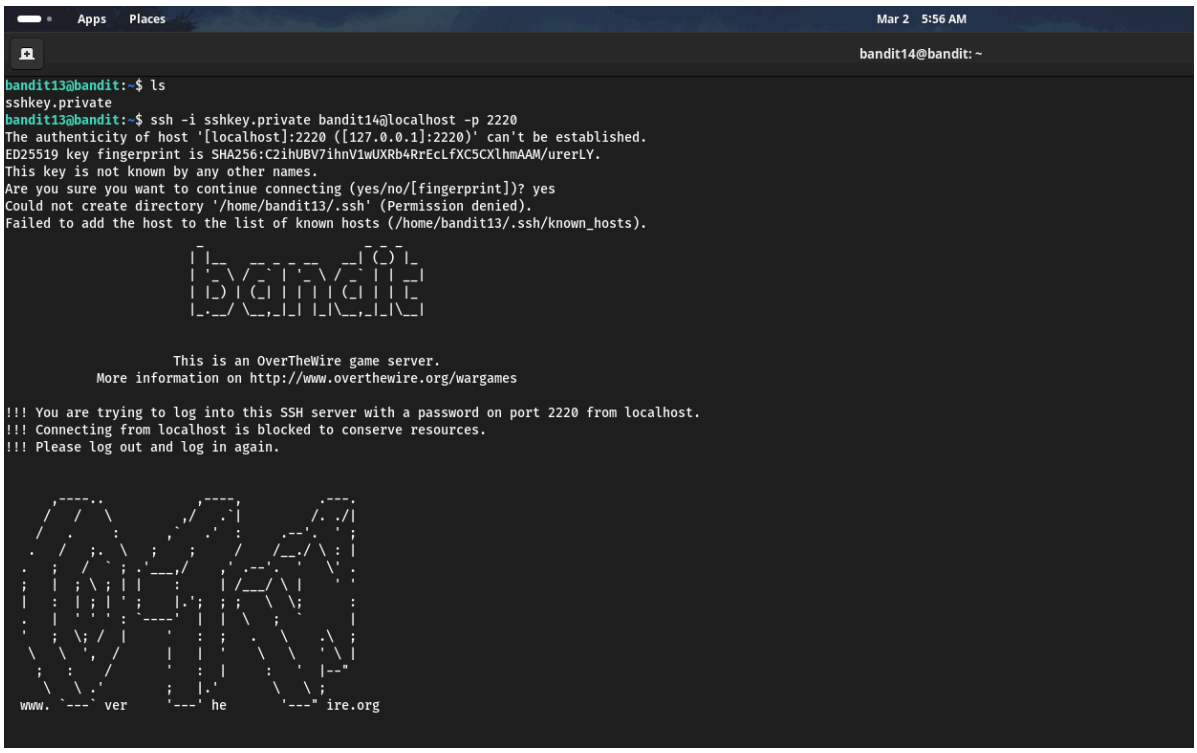
- `cat /etc/bandit_pass/bandit14` : Display the password for Level 14.

Screenshots

1. **SSH Connection**:



2. **Using the Private SSH Key**:



3. **Retrieving the Password**:

```
bandit14@bandit:~$ cat  /etc/bandit_pass/bandit14
MU4VWeTyJk8ROof1qqmcBPaLh7lDCPvS
bandit14@bandit:~$
```

Writeup for Bandit Level 14 → Level 15

Title: Bandit Level 14 - Submitting the Current Password to a Local Port

---

Introduction

Bandit Level 14 requires users to submit the current level's password to a specific port on `localhost` to retrieve the password for the next level. This writeup documents the steps taken to complete Level 14 and retrieve the password for Level 15.

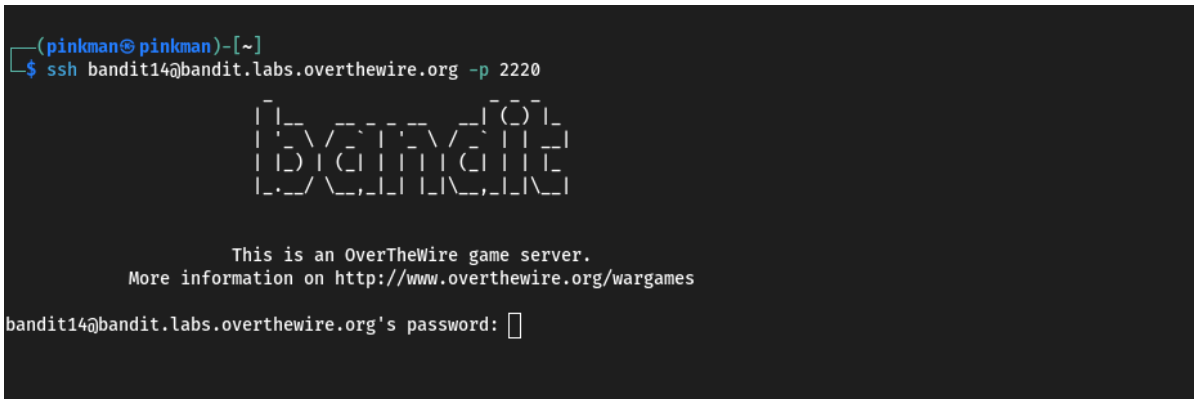
Level Goal

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

Methodology

1. **Connect to the Server Using SSH**:
   - Open a terminal and use the `ssh` command to connect to the server.
   - The command used is:

     `ssh bandit14@bandit.labs.overthewire.org -p 2220`
   - When prompted, enter the password retrieved from Level 13: `NU4VWeryJk8Roof1qqmcBPaLh7lDCPvS`.



2. **Access the Server**:
   - After successfully logging in, you will be in the home directory of the `bandit14` user.
3. **Submit the Current Password to Port 30000**:
   - Use the `nc` (netcat) command to connect to `localhost` on port `30000` and submit the current password:

     `echo "NU4VWeryJk8Roof1qqmcBPaLh7lDCPvS" | nc localhost 30000`
   - Alternatively, you can use:

     `nc localhost 30000`

     Then, manually type the password and press Enter.
4. **Retrieve the Password for Level 15**:

- After submitting the password, the server will respond with the password for Level 15.

```
bandit14@bandit:~$ ls
bandit14@bandit:~$ nc localhost 30000
MU4VWeTyJk8ROof1qqmcBPaLh7lDCPvS
Correct!
8xCjnmgoKbGLhHFAZlGE5Tmu4M2tKJQo

^C
bandit14@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```

Findings/Results

- The password for Level 15 is: `BxCjnmgokbGLhhFAZ1GE5Tmu4M2tKJQo`

Discussion/Analysis

- Level 14 introduces the concept of network communication with a local service using the `nc` command. The task involves sending the current password to a specific port to receive the next password.

- This level emphasizes the importance of understanding basic network communication and using tools like `nc` to interact with services.

Conclusion

- Successfully logged into the Bandit game server as `bandit14`.
- Submitted the current password to port `30000` on `localhost` using the `nc` command.
- Retrieved the password for Level 15 from the server's response.
- This level reinforces the importance of understanding network communication and using tools to interact with services.

Commands Used

- `ssh bandit14@bandit.labs.overthewire.org -p 2220`: Connect to the server via SSH.
- `echo "NU4VWeryJk8Roof1qqmcBPaLh7lDCPvS" | nc localhost 30000`: Submit the current password to port `30000` on `localhost`.

Screenshots

1. **SSH Connection**:

```
┌──(pinkman㉿pinkman)-[~]
└─$ ssh bandit14@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


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

bandit14@bandit.labs.overthewire.org's password:
```

2. **Retrieving the Password**:

```
bandit14@bandit:~$ ls
bandit14@bandit:~$ nc localhost 30000
MU4VWeTyJk8ROof1qqmcBPaLh7lDCPvS
Correct!
8xCjnmgoKbGLhHFAZlGE5Tmu4M2tKJQo

^C
bandit14@bandit:~$ exit
logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```

Writeup for Bandit Level 15 → Level 16

Title: Bandit Level 15 - Submitting the Current Password to a Local Port Using SSL/TLS

Introduction

Bandit Level 15 requires users to submit the current level's password to a specific port on `localhost` using SSL/TLS encryption to retrieve the password for the next level. This writeup documents the steps taken to complete Level 15 and retrieve the password for Level 16.

Level Goal

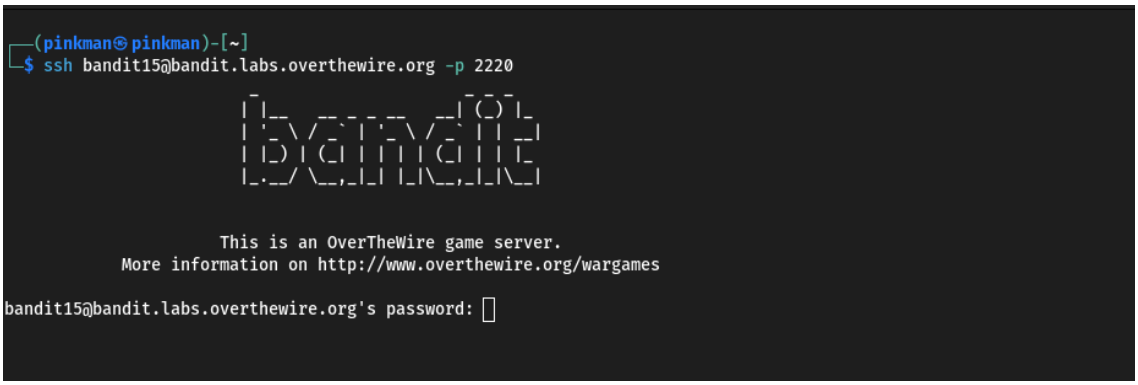
- The password for the next level can be retrieved by submitting the password of the current level to port `30001` on `localhost` using SSL/TLS encryption.

Methodology

1. **Connect to the Server Using SSH**:
   - Open a terminal and use the `ssh` command to connect to the server.
   - The command used is:

     `ssh` `bandit15@bandit.labs.overthewire.org` `-p 2220`
   - When prompted, enter the password retrieved from Level 14:

     `BxCjnmgokbGLhhFAZ1GE5Tmu4M2tKJQo`.



2. **Access the Server**:
   - After successfully logging in, you will be in the home directory of the `bandit15` user.
3. **Submit the Current Password to Port 30001 Using SSL/TLS**:
   - Use the `openssl s_client` command to connect to `localhost` on port `30001` using SSL/TLS:

     `openssl s_client -connect localhost:30001`
   - After establishing the connection, submit the current password:

     `BxCjnmgokbGLhhFAZ1GE5Tmu4M2tKJQo`
   - The server will respond with the password for Level 16.

```
bandit15@bandit:~$ openssl s_client -connect localhost:30001
CONNECTED(00000003)
Can't use SSL_get_servername
depth=0 CN = SnakeOil
verify error:num=18:self-signed certificate
verify return:1
depth=0 CN = SnakeOil
verify return:1
---
Certificate chain
 0 s:CN = SnakeOil
   i:CN = SnakeOil
   a:PKEY: rsaEncryption, 4096 (bit); sigalg: RSA-SHA256
   v:NotBefore: Jun 10 03:59:50 2024 GMT; NotAfter: Jun  8 03:59:50 2034 GMT
---
```

```
    Start Time: 1740926510
    Timeout   : 7200 (sec)
    Verify return code: 18 (self-signed certificate)
    Extended master secret: no
    Max Early Data: 0
---
read R BLOCK
8xCjnmgoKbGLhHFAZlGE5Tmu4M2tKJQo
Correct!
kSkvUpMQ7lBYyCM4GBPvCvT1BfWRy0Dx

closed
bandit15@bandit:~$ logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$ 
```

Findings/Results

- The password for Level 16 is: `kSkvUpMQ7LBYyCM4GBPvCVT1BfWRy0Dx`

Discussion/Analysis

- Level 15 introduces the concept of secure network communication using SSL/TLS. The task involves using the `openssl s_client` command to establish a secure connection and submit the current password.
- This level emphasizes the importance of understanding secure communication protocols and using tools like `openssl` to interact with secure services.

Conclusion

- Successfully logged into the Bandit game server as `bandit15`.
- Established a secure connection to port `30001` on `localhost` using the `openssl s_client` command.
- Submitted the current password and retrieved the password for Level 16 from the server's response.
- This level reinforces the importance of understanding secure communication and using appropriate tools to interact with secure services.

Commands Used

- `ssh bandit15@bandit.labs.overthewire.org -p 2220`: Connect to the server via SSH.
- `openssl s_client -connect localhost:30001`: Establish a secure connection to port `30001` on `localhost`.
- Submit the current password: `BxCjnmgokbGLhhFAZ1GE5Tmu4M2tKJQo`.

Screenshots

1. **SSH Connection**:

```
┌──(pinkman㉿pinkman)-[~]
└─$ ssh bandit15@bandit.labs.overthewire.org -p 2220
                         _                     _ _ _
                        | |__   __ _ _ __   __| (_) |_
                        | '_ \ / _` | '_ \ / _` | | __|
                        | |_) | (_| | | | | (_| | | |_
                        |_.__/ \__,_|_| |_|\__,_|_|\__|


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

bandit15@bandit.labs.overthewire.org's password:
```

2. **Establishing SSL/TLS Connection**:

```
bandit15@bandit:~$ openssl s_client -connect localhost:30001
CONNECTED(00000003)
Can't use SSL_get_servername
depth=0 CN = SnakeOil
verify error:num=18:self-signed certificate
verify return:1
depth=0 CN = SnakeOil
verify return:1
---
Certificate chain
 0 s:CN = SnakeOil
   i:CN = SnakeOil
   a:PKEY: rsaEncryption, 4096 (bit); sigalg: RSA-SHA256
   v:NotBefore: Jun 10 03:59:50 2024 GMT; NotAfter: Jun  8 03:59:50 2034 GMT
---
```

3. **Retrieving the Password**:

```
    Start Time: 1740926510
    Timeout   : 7200 (sec)
    Verify return code: 18 (self-signed certificate)
    Extended master secret: no
    Max Early Data: 0
---
read R BLOCK
8xCjnmgoKbGLhHFAZlGE5Tmu4M2tKJQo
Correct!
kSkvUpMQ7lBYyCM4GBPvCvT1BfWRy0Dx

closed
bandit15@bandit:~$ logout
Connection to bandit.labs.overthewire.org closed.

┌──(pinkman㉿pinkman)-[~]
└─$
```