Team Name: **HackHunters**

Team Members: **Haria Dhruti Jagdish**

Gala Dharmik Vimal

Reports:

Krypton

Level 0 → Level 1:

Objective:

The objective of **Krypton Level 0** is to decode a Base64-encoded password and use it to log in to the system with the username **krypton1** using SSH on port 2231. Upon logging in, the user can access files for the next levels in the /krypton/ directory.

Step-by-Step Solution:

1. **Base64 Decoding:**
   - **Tools used:** base64
   - **Action:** The password is provided as a Base64-encoded string: S1JZUFRPTklTR1JFQVQ=.
   - **Command:**

   -
     ```
     echo "S1JZUFRPTklTR1JFQVQ=" | base64 --decode
     ```
   - **Purpose:** To decode the Base64 string and obtain the password for the next step.
2. **Logging in via SSH:**
   - **Tools used:** ssh
   - **Action:** Use the decoded password to log in to the system with SSH.
   - **Command:**

   -
     ```
     ssh krypton1@krypton.labs.overthewire.org -p 2231
     ```

  - **Purpose:** To log in to the **krypton1** account on the server at the specified address and port using the decoded password.

3. **Accessing Files for Further Levels:**
   - **Tools used:** ls, cd
   - **Action:** After successfully logging in, navigate to the /krypton/ directory to access the files for other levels.
   - **Command:**
   - ls /krypton/
   - **Purpose:** To list the files that are part of the next levels in the **Krypton** wargame series.

Tools Used:

- **base64** - To decode the Base64-encoded password.
- **ssh** - To log in to the **krypton1** account on the remote server.
- **ls** - To list files in the /krypton/ directory.

Logic Behind the Solution:

The challenge involves decoding a password encoded in Base64 and using that password to log in to the remote server via SSH. Once logged in, the player can access the files for the subsequent levels by navigating to the /krypton/ directory.

Summary:

- **Decoded the Base64 string** to get the password.
- **Logged into the server** using the decoded password and the **krypton1** username via SSH on port 2231.
- **Accessed files** in the /krypton/ directory to proceed to the next levels.

This is how **Level 0** is solved in the **Krypton** wargame.

Level 1 → Level 2 :

Goal:

- Find the password for **Krypton Level 2**.
- The password is stored in the file krypton2.
- It is encrypted with a **simple rotation cipher** (ROT13 or similar).

Tools Used:

- cat — to display file contents.
- tr — to perform rotation cipher decoding.
- **Knowledge of ROT13 cipher**.

Step-by-Step Solution:

1. **Log in to Krypton1**
   You should already be logged into krypton1.
   If not:
2. ssh krypton1@krypton.labs.overthewire.org -p 2231
3. **List the available files**
   To check files present:
4. ls
5. **View the contents of the krypton2 file**
   To see what the encrypted password looks like:
6. cat krypton2
7. **Understand the encryption**
   - It is mentioned that it is a "simple rotation."
   - Most basic rotation cipher = **ROT13** (rotate letters by 13 positions).
8. **Decode the text using tr**
   To decode ROT13:
9. cat krypton2 | tr 'A-Za-z' 'N-ZA-Mn-za-m'
10. **Get the password**
    After decoding, the password for **Krypton Level 2** is revealed.

Logic Behind the Solution:

- ROT13 shifts each letter 13 places in the alphabet.
- Using the tr command with proper mapping allows quick decryption.
- Since the file structure retained spaces (not grouped 5-by-5), it made it easier to spot words after decoding.

Short List of Commands:

```
ssh krypton1@krypton.labs.overthewire.org -p 2231

ls

cat krypton2

cat krypton2 | tr 'A-Za-z' 'N-ZA-Mn-za-m'
```

✅ **Level 1 Solved! Password for Krypton2 obtained.**

Level 2 → Level 3:

🔵 **Goal:**
Find the password hidden in the file krypton3, encrypted using a **Caesar Cipher**.
You are provided with a **keyfile** and an **encrypt binary** to help you.

🛠️ **Tools Used:**

- cat
- mktemp
- cd
- ln -s
- chmod
- strings
- tr
- Basic understanding of Caesar cipher and rotation logic

🧠 **Logic Behind the Solution:**

- krypton3 file contains ciphertext encrypted with a Caesar cipher.
- We **do not know** the shift amount directly.
- But we can use /krypton/krypton2/encrypt to **encrypt** our known text and **compare** results.
- Since Caesar cipher just shifts letters, **we can decrypt** by trying shifts or using patterns.
- After decrypting, the **password** will be visible.

🪜 **Step-by-Step Solution:**

1. **Login** to Krypton Level 2:
2. ssh krypton2@krypton.labs.overthewire.org -p 2231
3. **View available files:**
4. ls
5. **Read the encrypted password:**
6. cat krypton3
7. **Create a temporary working directory:**
8. mktemp -d
9. **Move into that directory:**
10. cd /tmp/<generated_folder>
11. **Create a symbolic link to the keyfile:**
12. ln -s /krypton/krypton2/keyfile.dat
13. **Set directory permissions (important for encrypt binary access):**
14. chmod 777 .
15. **Use the encrypt program to encrypt some known text:**
16. /krypton/krypton2/encrypt /etc/issue
17. **Analyze the encrypted file (ciphertext) using strings or cat:**
18. strings ciphertext

19. **Based on the encryption pattern, guess the shift value.**

(You can manually decrypt using tr command once the shift is known.)

20. **Try to decode the password manually if necessary:**

21. echo "CIPHERTEXT" | tr 'A-Z' 'U-ZA-T'

(Adjust the tr shift range depending on encryption.)

22. **Retrieve the password for Level 3.**

⚡ Short List of Commands:

```
ssh krypton2@krypton.labs.overthewire.org -p 2231

ls

cat krypton3

mktemp -d

cd /tmp/<tempdir>

ln -s /krypton/krypton2/keyfile.dat

chmod 777 .

/krypton/krypton2/encrypt /etc/issue

strings ciphertext

echo "CIPHERTEXT" | tr 'A-Z' 'U-ZA-T'
```

Level 3 → Level 4 :

🎯 Goal

- Find the password for **krypton4**.
- Password is hidden inside the file named krypton4.
- Help is available: 3 additional intercepted encrypted messages (found1, found2, found3) were provided.
- All encrypted messages use the **same substitution key**.
- All plaintexts are in **American English**.

🛠 Tools Used

- ssh (for connecting to server)
- cat (to view file contents)
- freq / manual **frequency analysis** (letter count)
- Online tools or Python script for solving **simple substitution cipher**.

🧠 Logic Behind the Solution

- A simple substitution cipher replaces one letter with another.
- Since **multiple ciphertexts** were available (krypton4, found1, found2, found3), it allowed **frequency analysis**.
- In English, certain letters like E, T, A, O, N occur more frequently.
- By comparing the most common letters in the ciphertext to typical English frequencies, the substitution could be guessed.
- Once the substitution mapping was found, decrypt krypton4 and get the password.

🪜 Step-by-Step Solution

1. **Connect** to the Krypton server:
2. ssh krypton3@krypton.labs.overthewire.org -p 2231
3. **List files** to see what's available:
4. ls
5. **View the contents** of all files:
6. cat krypton4
7. cat found1
8. cat found2
9. cat found3
10. **Combine all encrypted files** to get a larger sample for frequency analysis:
11. cat krypton4 found1 found2 found3 > all_found
12. **Analyze letter frequencies manually** (or use a small script / online tool).

13. **Start decoding** using the frequency clues and adjust the mapping by guessing common English words like **"the"**, **"and"**, etc.
14. **Decrypt krypton4** fully using the mapping.
15. **Get the password** from the decrypted text.
16. **Login to krypton4** with the found password.

🔥 Short List of Commands

```
ssh krypton3@krypton.labs.overthewire.org -p 2231

ls

cat krypton4

cat found1

cat found2

cat found3

cat krypton4 found1 found2 found3 > all_found

cat all_found

# Frequency analysis manually or using a small script

# Decrypt manually using substitution guesses
```

Level 4 → Level 5:

🎯 Goal

- Find the password for **krypton5**.
- Password is inside the file krypton5.
- The password is **encrypted using a Vigenère cipher**.
- **Key length** is known: **6 characters**.
- Two intercepted English texts are provided for analysis.

🛠 Tools Used

- ssh (to connect to the server)

- ls, cat (to explore and read files)
- Manual analysis and frequency analysis (or online Vigenère decryption tools)
- Knowledge of Vigenère Cipher basics.

🧠 Logic Behind the Solution

- Vigenère Cipher shifts each letter based on a repeating key.
- Knowing the **key length (6)** is a major advantage.
- Strategy:
  - Group letters according to their position in the key (every 6th letter belongs to the same group).
  - Perform **frequency analysis** on each group separately.
  - Assume 'E' is the most frequent letter in English text → use that to guess the shift for each group.
  - Deduce the key by aligning the most frequent letters back to 'E'.
  - Use the recovered key to decrypt the file krypton5.

🪜 Step-by-Step Solution

1. **Connect** to the Krypton server:
2. ssh krypton4@krypton.labs.overthewire.org -p 2231
3. **List the available files**:
4. ls
5. **View contents** of all intercepted files and the encrypted password:
6. cat krypton5
7. cat found1
8. cat found2
9. **Perform manual grouping** of letters based on key length 6:
   - For example, letters at position 1, 7, 13... belong to group 1, and so on.
10. **Analyze each group separately** using frequency analysis.

11. **Estimate the key** by assuming the most common letter in English is 'E'.
12. **Decrypt the krypton5 ciphertext** using the discovered Vigenère key.
13. **Extract the password** and login to **krypton5**.

🔥 Short List of Commands

```
ssh krypton4@krypton.labs.overthewire.org -p 2231

ls

cat krypton5

cat found1

cat found2

# Manual analysis to break Vigenère cipher

# Decrypt krypton5 using recovered key
```

✅ End of Krypton Level 4 → Level 5 Report!

Level 5 → Level 6:

🎯 Goal:

Find the password hidden inside the file by decrypting a Vigenère cipher with an **unknown key length**.

🛠️ Tools Used:

- ssh (for login)
- ls (to list files)
- cat (to read encrypted data)
- Online tools or manual techniques:
  - **Kasiski Examination** (to guess the key length)
  - **Frequency Analysis** (to guess the key)
  - **Vigenère Decryptor** (manual or online)

🧠 Logic Behind the Solution:

- The encrypted file is protected by a **Vigenère cipher**.
- Unlike the previous level, **key length is unknown**.
- First step: **analyze repeating patterns** to estimate the key length (Kasiski method).
- Second step: once key length is guessed, perform **frequency analysis** on each part to guess the key characters.
- Final step: **decrypt** the file with the found key and extract the password.

📝 Step-by-Step Solution:

1. **Connect to the server**:
2. ssh krypton5@krypton.labs.overthewire.org -p 2231
3. **Check available files**:
4. ls
5. **View the encrypted file**:
6. cat krypton6
7. **Analyze the ciphertext manually**:
    - Notice patterns and repeated sequences.
    - Use the **Kasiski method** to guess key length.
    - Based on repeated patterns distance, guess probable key length (like 7, 8, etc.)
8. **Frequency analysis**:
    - Break the ciphertext into groups based on guessed key length.
    - For each group, analyze the most frequent letter and assume it represents 'E' (the most common letter in English).
9. **Guess the key manually**.
10. **Decrypt** the ciphertext using the derived key to retrieve the password for krypton6.

✅ Password for Krypton Level 6 is found!

Short list of commands:

```
ssh krypton5@krypton.labs.overthewire.org -p 2231

ls

cat krypton6

# manual analysis: kasiski, frequency analysis, decrypt
```

✅ **End of short commands for Krypton Level 5 → Level 6!**

Level 6 → Level 7:

Goal

- Find the password to log into **krypton7**.
- Password is encrypted using a **weak stream cipher** and stored in the file.

Tools Used

- ssh (for login)
- ls, cat (to explore the files)
- hexdump, xxd (to view raw binary/hex data)
- encrypt6 binary (to encrypt custom plaintexts)
- XOR attack logic
- Basic hex editors

Step-by-Step Solution

1. **Login to krypton6**
2. ssh krypton6@krypton.labs.overthewire.org -p 2231
3. **List the directory contents**
4. ls
    - Files seen: keyfile.dat, encrypt6, krypton7

5. **Check what is readable**
    - keyfile.dat was **not readable**.
    - encrypt6 was **executable**.
6. **Use hexdump to inspect the password file**
7. hexdump -C krypton7
    - We see raw hex-encoded encrypted password.
8. **Understand encrypt6 usage**
    - encrypt6 encrypts any input given and produces ciphertext based on an internal weak PRNG (pseudo random number generator).
9. **Perform a known-plaintext attack**
    - Create controlled plaintext (like repeating characters, e.g., all "A"s).
    - Encrypt it using encrypt6 and capture the ciphertext output.

Example:

echo -n "A" > inputfile

./encrypt6 inputfile

hexdump -C ciphertext

10. **Analyze ciphertexts**
    - XOR known plaintext with ciphertext byte-by-byte to guess the keystream.
    - Once the keystream is partially known, XOR it with the ciphertext in krypton7 to recover plaintext.
11. **Extract password**
    - After XOR-ing correctly, retrieve the **plaintext password** from the decrypted bytes.
12. **Login to krypton7 using the password**.

Logic Behind the Solution

- The encryption uses a **weak stream cipher** with a **predictable random number generator**.

- Since stream ciphers encrypt data by XOR-ing plaintext with a keystream, if we can control the plaintext, we can extract parts of the keystream.
- Once the keystream is recovered (or guessed sufficiently), we can XOR it with the ciphertext (krypton7) to recover the original password.
- **Known-plaintext attack** is very powerful against weak stream ciphers.

✅ **Krypton Level 6 → Level 7 Completed!**