# Cybersecurity Wargame Internship Task

## Intern Name: Isha Adangale

## Commands

### Level 0 -> Level 1

echo "S1JZUFRPTklTQ1JFQVQ=" | base64 -d

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

cd /krypton/krypton1

cat README

cat krypton2"[NOPQRSTUVWXYZABCDEFGHIJKLM cat krypton2 | tr "[ABCDEFGHIJKLMNOPQRSTUVWXYZ]"]"

### Level 1 -> Level 2

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

cat krypton2

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

### Level 2 -> Level 3

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

mktemp -d

cd /tmp/tmp.<random_string>

ln -s /krypton/krypton2/keyfile.dat

ls

chmod 777 .

ln -s /krypton/krypton2/encrypt /etc/issue

mktemp -d

cd /tmp/tmp.<random_string>

ln -s /krypton/krypton2/keyfile.dat

ls

chmod 777 .

ls

cat /etc/issue

/krypton/krypton2/encrypt /etc/issue

ls

cat ciphertext

touch ptext

nano ptext

cat ptext

/krypton/krypton2/encrypt ptext

ls

cat ciphertext

cat /krypton/krypton2/krypton3

cat /krypton/krypton2/krypton3 | tr "[MNOPQRSTUVWXYZABCDEFGHIJKL]" "[A-Z]"

## Level 3 -> Level 4

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

ls -l

file krypton4

cat krypton4

strings krypton4

cat krypton4 | tr "JDSQBKVIWGYUNCXM" "THEAOLVDNPSRIFU"

## Level 4 -> Level 5

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

ls -l

file krypton5

cat krypton5

strings krypton5

python3 vignere_decoder.py /krypton/krypton4/krypton5 FREKEY

## Level 5 -> Level 6

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

ls -l

file krypton6

cat krypton6

strings krypton6

python3 vignere_decoder.py /krypton/krypton5/krypton6 KEYLENGTH

## Level 6 -> Level 7

ssh krypton6@krypton.labs.overthewire.org -p 2231

ls -l

file krypton7

cat a.txt

./krypton6/encrypt6 a.txt cipher_a.txt

cat cipher_a.txt

cat cipher

ls

python3 vignere_decoder.py /krypton/krypton6/krypton7
EICTDGYIYZKTHNSIRFXYCPFUEOCKRN

---

**Krypton Report**

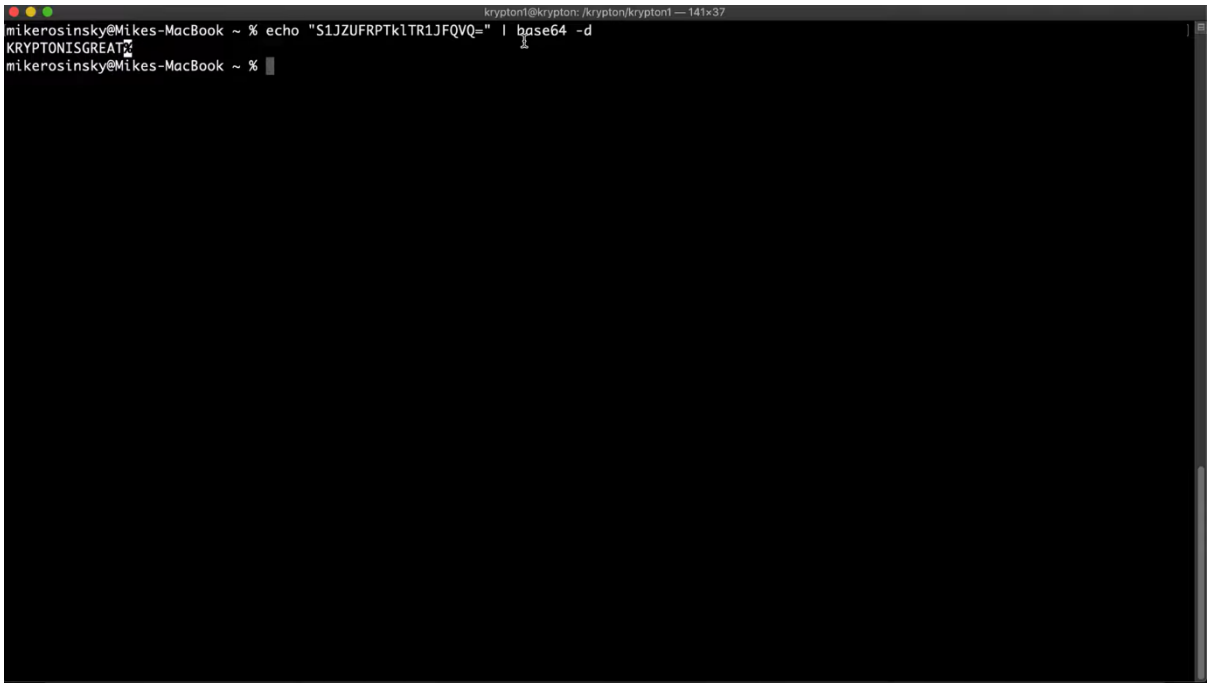
**Intern Name: Isha Adangale**

---

## Level 0 → Level 1

**Steps :**

- Decoded a base64-encoded string to obtain an SSH connection hint.




- Connected to the server using SSH.
- Navigated to the challenge directory and read the provided instructions.
- Found the encrypted password file and identified it was encoded using a simple letter shift (Caesar cipher).
- Decrypted it by applying a shift of 13 (ROT13) to recover the password.

**Tools used:**

- base64
- ssh
- tr command (text replacement)

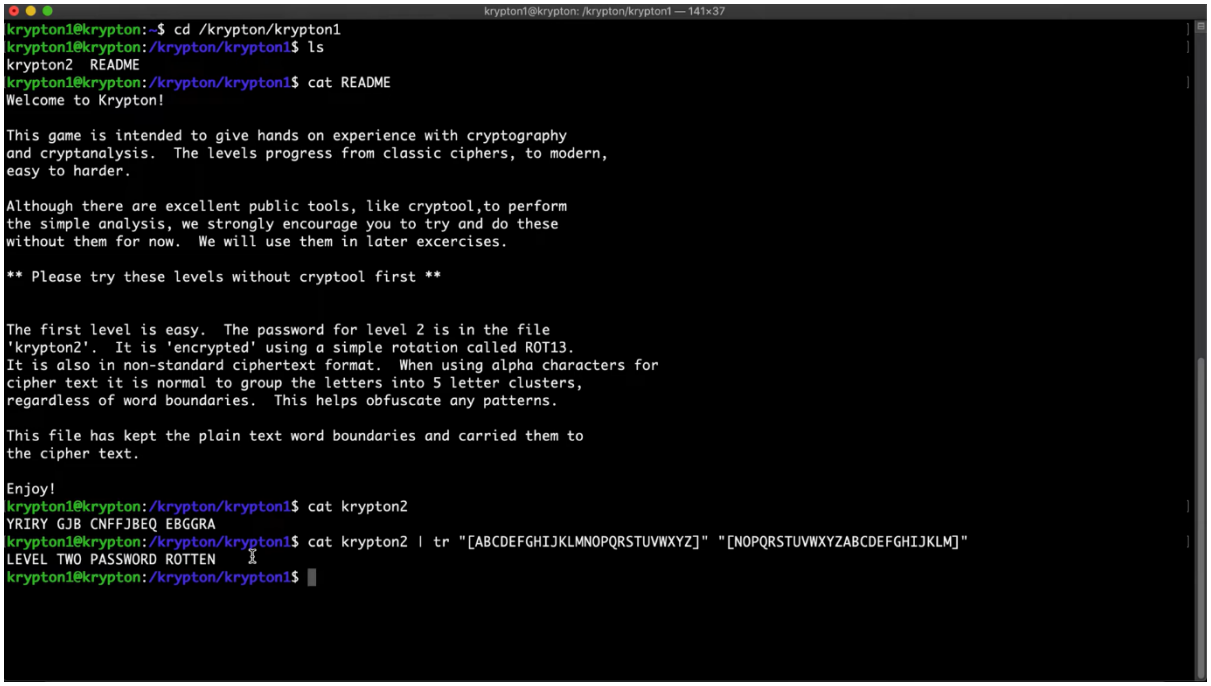
## Level 1 → Level 2

**Steps :**

- Logged in to the server with the new password.
- Retrieved the new password file.
- Recognized the encryption was again a ROT13 cipher.
- Decrypted the password.

**Tools used:**

- ssh
- tr command



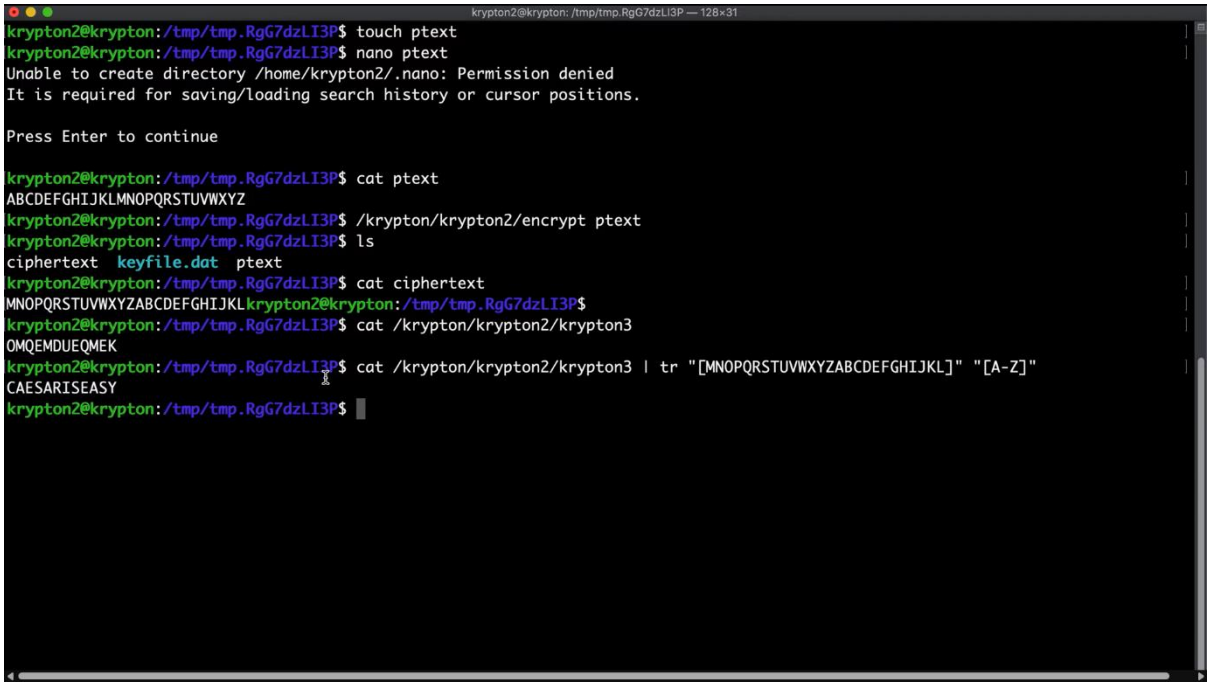
## Level 2 → Level 3

**Steps :**

- Logged into the server with the Level 2 password.
- Created a temporary directory and symbolic links to important files (encryption key and program).
- Manipulated file permissions to control file behavior.
- Encrypted known plaintexts to study the encryption.
- Analyzed the output to infer the encryption method.
- Decrypted the encrypted password using a Caesar cipher based on the learned behavior.

**Tools used:**

- mktemp
- ln (for symbolic links)
- chmod
- cat
- tr



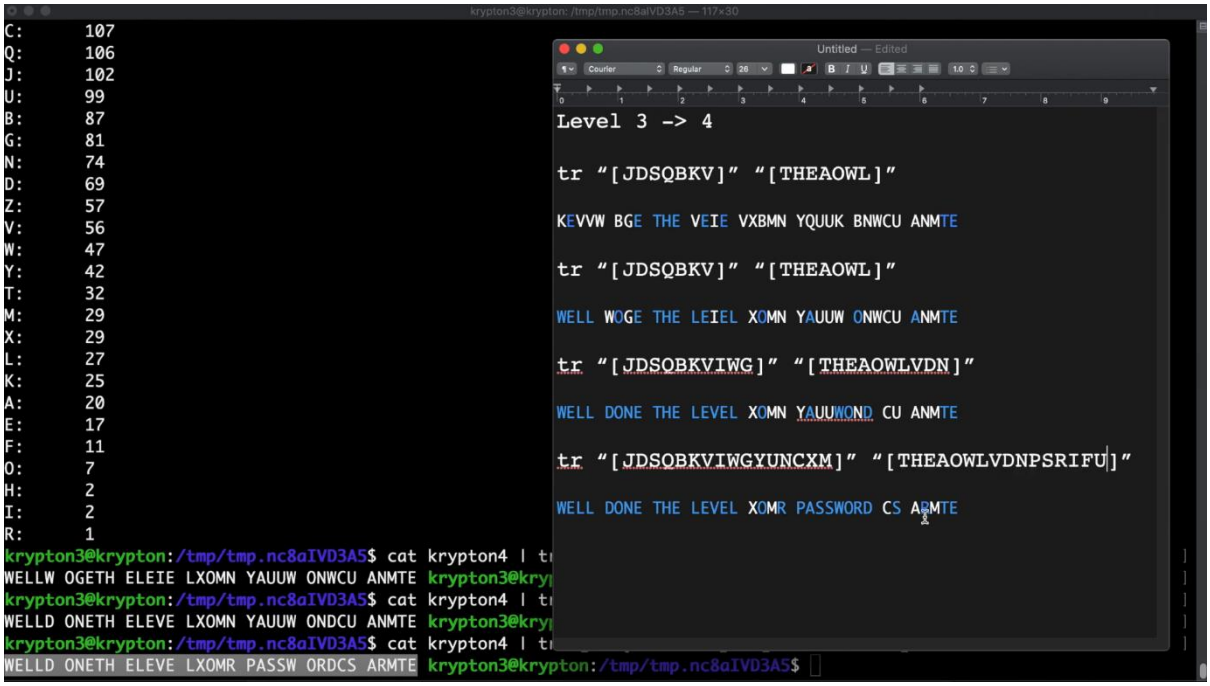
## Level 3 → Level 4

**Steps :**

- Logged into Level 3.
- Inspected the given file to check the file type and its contents.
- Extracted readable text using string extraction.
- Recognized a monoalphabetic substitution cipher.
- Built a mapping based on letter frequency and patterns to decrypt the password.

**Tools used:**

- strings
- tr
- Manual cipher analysis



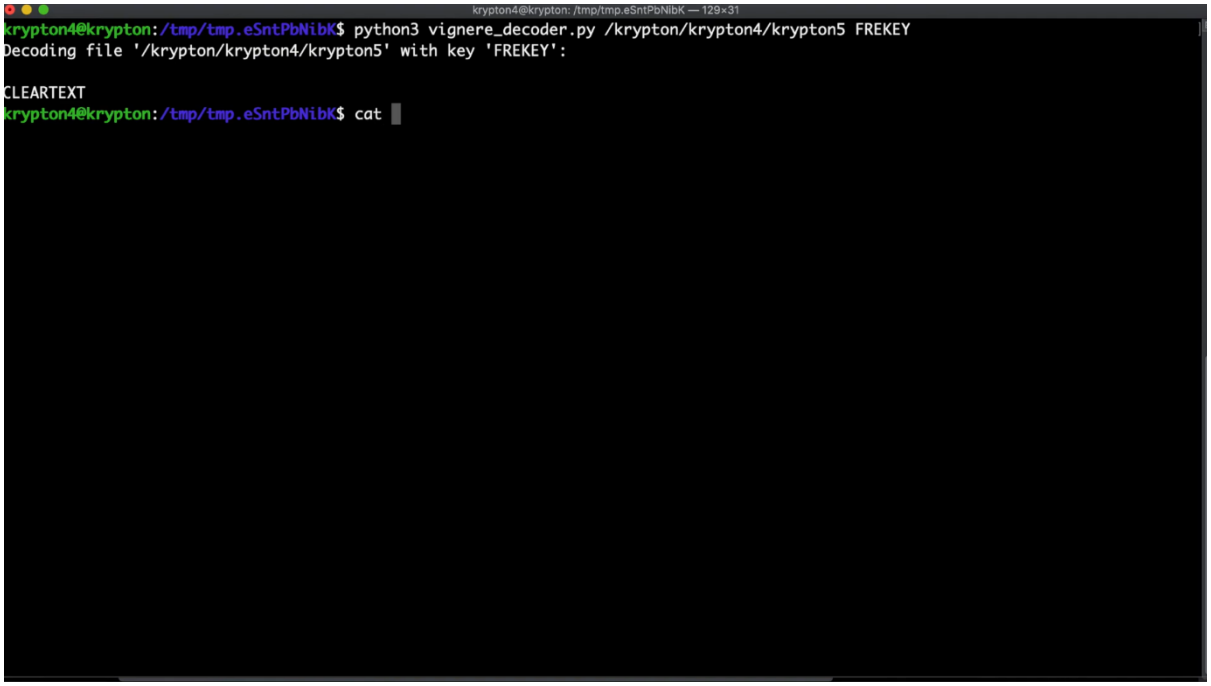
## Level 4 → Level 5

**Steps :**

- Logged into Level 4.
- Opened and analyzed the encrypted password file.
- Identified that the cipher was Vigenère.
- Used a Python script to automate Vigenère decryption with a given key.

**Tools used:**

- strings
- python3 (custom Vigenère decryption script)



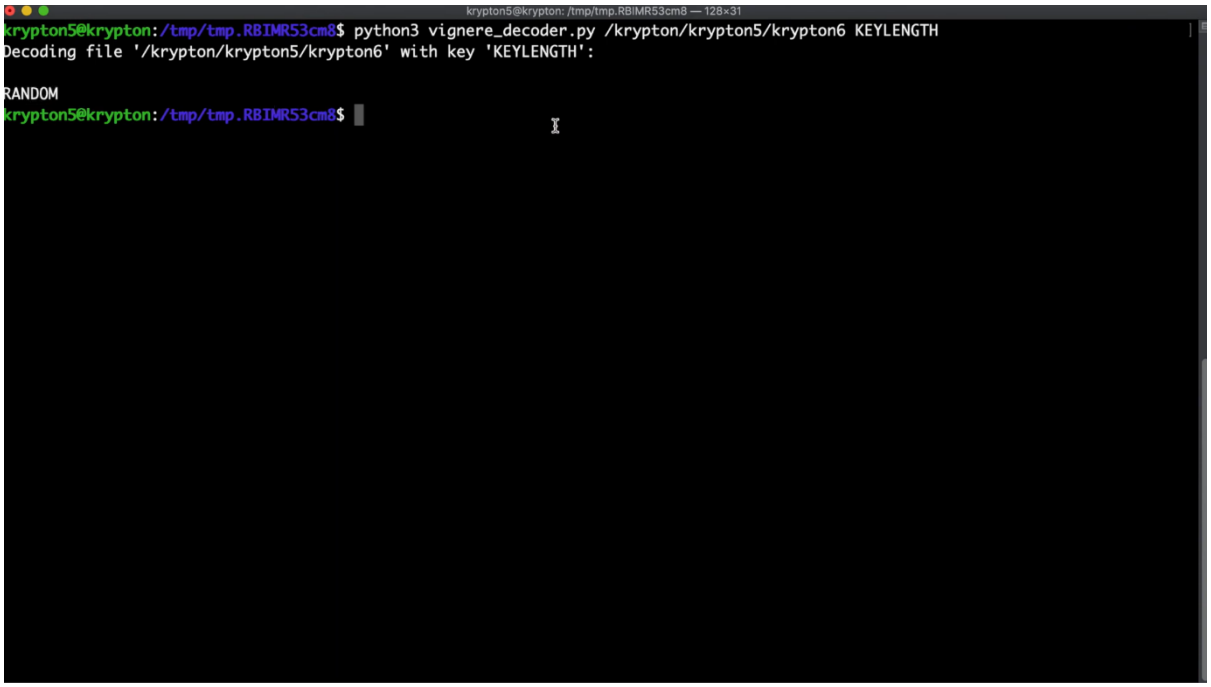
## Level 5 → Level 6

**Steps :**

- Connected to Level 5.
- Inspected the encrypted file.
- Realized Vigenère cipher was used again, but without knowing the key length.
- Performed key length analysis by testing different possibilities.
- Used the decryption script with the correct key length to retrieve the password.

**Tools used:**

- strings
- python3 (modified Vigenère decryption script)



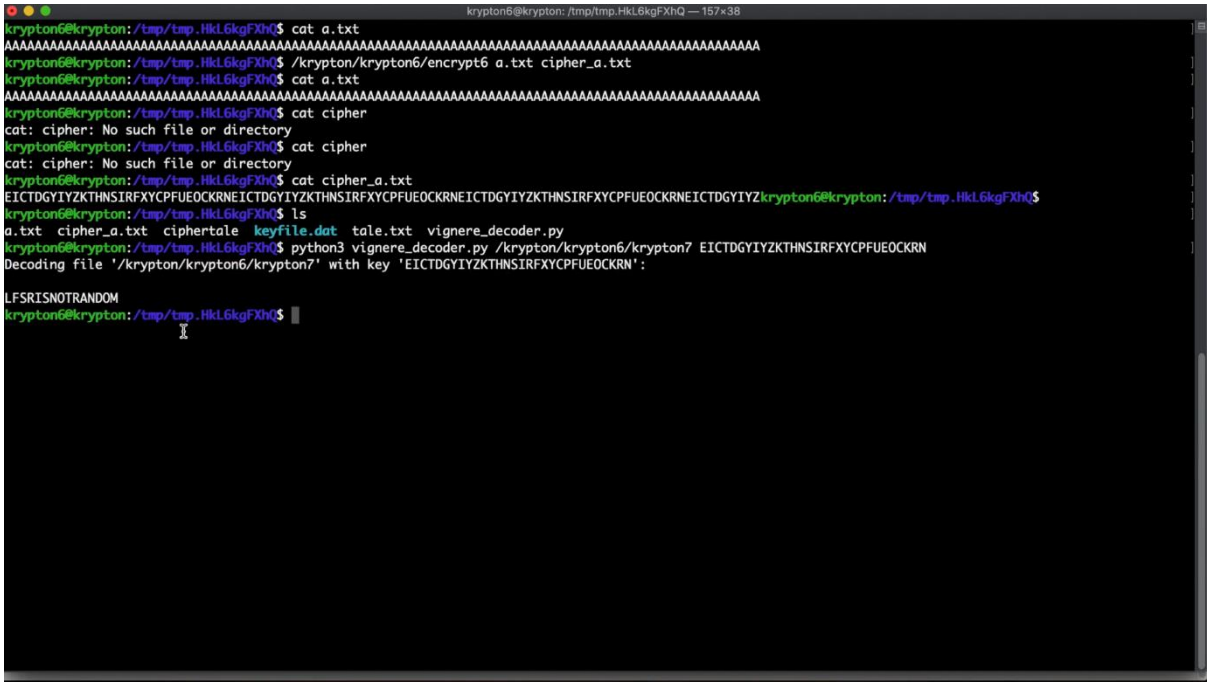
## Level 6 → Level 7

**Steps :**

- Logged into Level 6.
- Found a new encryption tool and the encrypted password file.
- Created known plaintext files and encrypted them to analyze encryption behavior.
- Observed how the encryption tool modified the plaintext.
- Reconstructed the full key based on ciphertext analysis.
- Decrypted the final ciphertext using the full key.

**Tools used:**

- encrypt6 program
- Manual file manipulation
- python3 (Vigenère decryption)



**Level 7 → Level 8**

**Steps:**

- Logged in and read final congratulation message (no cracking needed).

Krypton series completed successfully