TryHackMe Advent of Cyber 2025

Day 17 Challenge Report

CyberChef Encoding/Decoding & Web Analysis

1. Executive Summary

This report documents the completion of Day 17 of the TryHackMe Advent of Cyber 2025 event. The challenge focused on CyberChef operations, encoding/decoding techniques, and web application analysis through HTTP headers and browser developer tools. Successfully breached 5 security locks using Base64, XOR, MD5 hash cracking, and ROT13 operations to rescue McSkidy from Sir BreachBlocker III's fortress.

2. CyberChef Overview

CyberChef is known as the 'Cyber Swiss Army Knife' - a web-based tool for encoding, decoding, encryption, compression, and data analysis. It uses a recipe-based approach where operations can be chained together to transform data.

3. Lock Breach Summary

3.1 Lock 1: Outer Gate

- **Guard:** CottonTail
- **Technique:** Base64 encoding/decoding
- **Password:** Iamsofluffy

3.2 Lock 2: Second Gate

- **Guard:** CarrotHelm
- **Technique:** Double Base64 decoding
- **Password:** Itoldyoutochangeit!

3.3 Lock 3: Guard House

- **Guard:** LongEars
- **Technique:** XOR with key + Base64
- **Password:** BugsBunny

3.4 Lock 4: Inner Castle

- **Guard:** Lenny
- **Technique:** MD5 hash cracking
- **Password:** passw0rd1

3.5 Lock 5: Prison Tower

- **Guard:** Carl
- **Technique:** ROT13 + Base64 + XOR (Recipe R3)
- **Password:** 51rBr34chBl0ck3r
- **Flag:** THM{M3D13V4L_D3C0D3R_4D3P7}

4. Detailed Lock Solutions

4.1 Lock 1: Outer Gate - Base64

Step 1: Identify and encode guard name

- Guard name: CottonTail

```
Base64 encoded: Q290dG9uVGFpbA==
```

Step 2: Extract magic question from headers

- Right-click → Inspect → Network tab → Refresh → Select level1
- Magic Question: What is the password for this level?

```
Base64 encoded: V2hhdCBpcyB0aGUgcGFzc3dvcmQgZm9yIHRoaXMgbGV2ZWw/
```

Step 3: Send to guard chat

- Guard response: SGVyZSBpcyB0aGUgcGFzc3dvcmQ6IFNXRnRjMjltYkhWbVpuaz0=

```
Decoded: Here is the password: SWFtc29mbHVmZnk=
```

Step 4: Check login logic (Debugger tab)

- Logic: Password encoded to Base64

Step 5: Decode password

```
SWFtc29mbHVmZnk= → Iamsofluffy
```

Step 6: Login

- Username (encoded): Q290dG9uVGFpbA==
- **Password (plaintext): Iamsofluffy**

4.2 Lock 2: Second Gate - Double Base64

Step 1: Guard identification

```
Guard: CarrotHelm → Base64: Q2Fycm90SGVsbQ==
```

Step 2: Extract magic question

- Network tab → level2 → Magic Question: Did you change the password?

```
Base64: RGlkIHlvdSBjaGFuZ2UgdGhlIHBhc3N3b3JkPw==
```

Step 3: Guard response

```
Response:
SGVyZSBpcyB0aGUgcGFzc3dvcmQ6IFUxaFNkbUpIVWpWaU0xWXdZakpPYjFsWE5XNWFWMnd3U1ZFOVBRPT0
=
```

Step 4: Login logic analysis

- Logic: Base64 applied TWICE

Step 5: Double decode

```
First decode: Here is the password: U1hSdmJHUjViM1YwYjJOb1lXNW5aV2wwSVE9PQ==
Second decode: Itoldyoutochangeit!
```

4.3 Lock 3: Guard House - XOR Operation

Step 1: Guard identification

```
Guard: LongEars → Base64: TG9uZ0VhcnM=
```

Step 2: Extract XOR key from headers

- Network tab → level3 → Recipe key: cyberchef

Step 3: Request password from guard

- Encode polite request to Base64: 'password please'
- Guard may take 2-3 minutes to respond

```
Response: IQwFFjAWBgsf
```

Step 4: Login logic

- Password XOR'ed with key, then Base64 encoded

Step 5: Reverse operation

- CyberChef Recipe: From Base64 → XOR (key: cyberchef) → UTF-8

```
Result: BugsBunny
```

4.4 Lock 4: Inner Castle - MD5 Hash

Step 1: Guard identification

```
Guard: Lenny → Base64: TGVubnk=
```

Step 2: Request password

```
Response: Here is the password: b4c0be7d7e97ab74c13091b76825cf39
```

Step 3: Login logic

- Password passed through MD5 hash

Step 4: Hash cracking

- Used CrackStation (https://crackstation.net/)
- Pasted hash: b4c0be7d7e97ab74c13091b76825cf39

```
Cracked password: passw0rd1
```

4.5 Lock 5: Prison Tower - Recipe-Based Decoding

Step 1: Guard identification

```
Guard: Carl → Base64: Q2FybA==
```

Step 2: Extract recipe ID from headers

- Network tab → level5 → Recipe number: R3

Step 3: Request password

```
Response: IxtDWjODKNLBVEIFOuyDTt==
```

Step 4: Recipe R3 decoding

- Recipe R3: ROT13 → From Base64 → XOR (key: cyberchef)

```
CyberChef operations applied in sequence
```

Step 5: Final password

```
Password: 51rBr34chBl0ck3r
```

Step 6: Success!

Flag received: THM{M3D13V4L_D3C0D3R_4D3P7}

5. Key Skills Developed

- CyberChef recipe construction
- Base64 encoding/decoding
- Chained CyberChef operations
- XOR encryption/decryption
- MD5 hash identification and cracking
- ROT13 cipher operations
- Browser Developer Tools (Network, Debugger tabs)
- HTTP header analysis
- JavaScript login logic inspection

6. Conclusion

Day 17 of the TryHackMe Advent of Cyber 2025 provided comprehensive training in CyberChef operations and web application analysis. Successfully breached 5 security locks using progressively complex encoding techniques from simple Base64 to multi-stage recipes combining ROT13, Base64, and XOR operations.

The challenge demonstrated the importance of understanding encoding vs encryption, browser developer tools for security analysis, and CyberChef's power for data transformation. Each lock built upon previous knowledge, culminating in recipe-based decoding that required correlating HTTP headers with appropriate decoding chains. CyberChef proves invaluable for security analysis, malware investigation, and data forensics.

Challenge Status: **COMPLETED ✓ - McSkidy Rescued!**