Forensic CTF Challenge Write-up

Challenge Name: Artifact Trail

Category: Forensics

Type: Multi-layered Forensic CTF (Archive + Steganography + Audio Forensics + QR Decoding)

Author: Karthik

Challenge Description

In "*X Marks It*", players are given a password-protected ZIP archive named `x_marks_it.zip`. Within it lies a multilayered treasure hunt that tests a wide range of forensic techniques — from decoding strings, uncovering hidden images, extracting embedded files with `binwalk`, decoding Morse audio, to finally recovering and scanning a QR code containing the flag.

Tools & Techniques Used

- 7z: For archive extraction
- strings: To find clues in files
- ls -la: For discovering hidden files
- gzip: For decompressing .gz files
- binwalk -e: To extract embedded files from images
- stegcracker / stegseek: To crack hidden content in audio
- Morse Code Decoder (online)
- cat: To read extracted content
- base64 -d: To decode the base64-encoded image
- QR code scanner: To retrieve the final flag

Steps to Solve

1. Extract Archive

```
7z x x_marks_it.zip
```

Prompted for a password. Use `strings x_marks_it.zip` to discover the password hint and extract it.

2. Explore Extracted Files

Found: `Captains_diary.txt`, `script`, and a hidden `.Lost_map.jpg.gz`.

```
gzip -d .Lost_map.jpg.gz
```

3. Extract Embedded Audio from Image

```
binwalk -e .Lost_map.jpg
```

Reveals `morse.wav` hidden inside the image.

4. Crack Stego Password from Audio

```
stegcracker morse.wav
```

Password found: **treasure**, which reveals `morse.wav.out` (a text file with clues).

5. Decode Morse Clue

Audio or output reveals:

```
BLACK-SAILS-NEVER-DIE
```

Key hint: **goldcoin**

6. Run Custom Binary Script

```
./script
```

When prompted:

```
Phrase: BLACK-SAILS-NEVER-DIE
Key: goldcoin
```

7. Extract Final Hidden File

The script reveals a base64 encoded image (e.g., in `pirate_treasure.txt`).

8. Scan the QR Code

Scan `final_qr.png` to retrieve the final flag.

🎯 Skills Tested

- Deep file system exploration (`ls -la`, hidden files)
- Password clue identification using `strings`
- Steganography in audio files
- Morse code interpretation
- Multi-layered embedded data extraction
- Scripting interaction and base64 decoding
- QR code recognition and scanning

🏴 Final Flag

> **PIRATE{Y0u_F0und_Me_4h0y!}**