

## CTF Report

**Full Name** : Suresh Babu B  
**Program** : HCS-Penetration Testing 1-Month Internship  
**Date** : 10-03-2025

---

### CHALLENGE-1

**Category:** Web

**Description:** A challenge focused on analyzing web application behavior and uncovering hidden files.

**Challenge Overview:**

In this challenge, you must explore the given web application and identify hidden files or directories that may contain valuable information. Pay close attention to common misconfigurations that can leak sensitive data.

**Steps for Finding the Flag:**

1. **Initial Reconnaissance:** Start by analyzing the website's structure and available resources. Look for common files that may provide insights.
2. **Checking robots.txt:** Access the robots.txt file by navigating to <https://lock-web-web.hackatronics.com/robots.txt>. This file often contains disallowed directories that might hold sensitive information.
3. **Exploring Restricted Directories:** Use the discovered paths to check for files that might expose data or hints.
4. **Flag Retrieval:** Once the flag is found, capture and document it for submission.

**Flag:** flag{V13w\_r0b0t5.txt\_c4n\_b3\_u53ful!!!}

### CHALLENGE-2

**Category:** Web

**Description:** A challenge that requires exploring a web application, navigating through different pages, and analyzing external resources to uncover hidden information.

**Challenge Overview:**

In this challenge, you will need to thoroughly explore the web application and analyze various elements to find the flag. Pay close attention to scripts, external resources, and hidden files that may contain useful information.

### Steps for Finding the Flag:

1. **Initial Reconnaissance:** Start by exploring <https://the-world-web.hackatronics.com> and navigate through its pages to understand the structure.
2. **Dashboard Exploration:** Go to <https://the-world-web.hackatronics.com/dashboard.html> and investigate its contents for any hints or vulnerabilities.
3. **Analyzing External Resources:** Examine external scripts like <https://kit.fontawesome.com/3bb29e5d19.js> to see if they leak any useful information.
4. **Finding the Flag:** Discover and access <https://the-world-web.hackatronics.com/secret.txt>, which contains an encoded Base64 flag.
5. **Decoding the Flag:** Convert the Base64-encoded text to reveal the final flag.

**Flag:** FLAG{Y0u\_hav3\_4xpl0reD\_th3\_W0rLd!}

### CHALLENGE-3

**Category:** Network Forensics

**Description:** A challenge focused on file forensics and data recovery techniques.

#### Challenge Overview:

In this challenge, you are provided with a corrupted PNG file. Your task is to analyze and repair the file to retrieve the hidden flag. Understanding file structures and utilizing forensic tools will be key to solving this challenge.

### Steps for Finding the Flag:

1. **File Analysis:** Inspect the corrupted PNG file using tools like binwalk, xxd, or file to determine the nature of the corruption.
2. **Header Examination:** Identify if the file's header is damaged or missing, as this is a common issue with corrupted PNG files.
3. **Repairing the File:** Use tools such as **EaseUS Data Recovery** or **PNG repair tools** to attempt reconstruction of the damaged file.
4. **Extracting the Flag:** Once repaired, open the file to reveal the flag embedded within.

**Flag:** flag{m3ss3d\_h3ad3r\$}



## CHALLENGE-4

**Category:** Reverse Engineering

**Description:** A challenge that involves analyzing an old project file, identifying hidden data, and decoding it using a simple cipher.

### Challenge Overview:

In this challenge, you are given an .aia project file that was created using **MIT App Inventor**. Your goal is to import the file into the appropriate environment, extract the hidden message, and decode it using a **ROT47 cipher** to retrieve the flag.

### Steps for Finding the Flag:

1. **Identifying the File Type:** The provided .aia file is recognized as a project file from **MIT App Inventor**.
2. **Importing the File:** Open **MIT App Inventor** (ai2.appinventor.mit.edu) and import the .aia file.
3. **Locating the Hidden Text:** Explore the project files, blocks, or assets to find the encoded text: 7=28LE\_\_0>F490C6GbCD?8N.
4. **Decoding the Text:** Use a **ROT47 decoder** (online tools like CyberChef or Python scripts) to decrypt the message.
5. **Retrieving the Flag:** After decoding, the flag is revealed as:

**Flag:** flag{t00\_much\_rev3rsng}

## CHALLENGE-5

**Category:** OSINT (Open Source Intelligence)

**Description:** A challenge that requires using open-source investigation techniques to track down hidden information on the internet.

### Challenge Overview:

In this challenge, you must uncover confidential secrets hidden by Mr. TrojanHunt. By utilizing search engines and online archives, you will piece together clues to retrieve the flag.

### Steps for Finding the Flag:

1. **Google Search:** Search for "Mr. TrojanHunt" on Google and examine the search results.
2. **Finding the Right Link:** Click on the **third** link in the search results, which leads to [https://archive.org/details/secret\\_202103](https://archive.org/details/secret_202103).



3. **Exploring the Archive:** Navigate through the archived page and inspect the text or available files.
4. **Retrieving the Flag:** Identify the hidden text containing the flag and document it for submission.

**Flag:** flag{Tr0j3nHunt\_t1m3\_tr4v3l}

## CHALLENGE-6

**Category:** Cryptography

**Description:** A challenge that requires recognizing and decoding an obscure esoteric cipher to reveal the hidden flag.

### Challenge Overview:

In this challenge, you are given an encoded sequence of symbols that appears to be an esoteric programming language. After analyzing its structure, you identify it as **ReverseFuck**, a variation of Brainfuck. Your task is to decode the message to retrieve the flag.

### Steps for Finding the Flag:

1. **Analyzing the Cipher:** The given string consists of `<>+-[]`, symbols, indicating that it belongs to the **Brainfuck family** of esoteric languages.
2. **Identifying the Language:** By searching for "**reversefuck interpreter**", you determine that it is encoded using **ReverseFuck**, a reversed execution variant of Brainfuck.
3. **Decoding the Message:** Use an **online ReverseFuck interpreter** (such as a Brainfuck variant decoder) or write a Python script to execute the code.
4. **Retrieving the Flag:** After decoding, the flag is revealed as:

**Flag:** flag{R3vers3ddd\_70\_g3t\_m3}

## CHALLENGE-7

**Category:** Cryptography

**Description:** A challenge that involves decoding an esoteric programming language to uncover the hidden flag.

### Challenge Overview:

In this challenge, you receive a recipe written in an esoteric language similar to Chef. Your task is to interpret and correct the code, run it, and extract the hidden Brainfuck code, which must then be decoded to reveal the flag.



\_\_\_\_\_

- [illegible]

**Flag:** flag{y0u\_40+\_s3rv3d!}