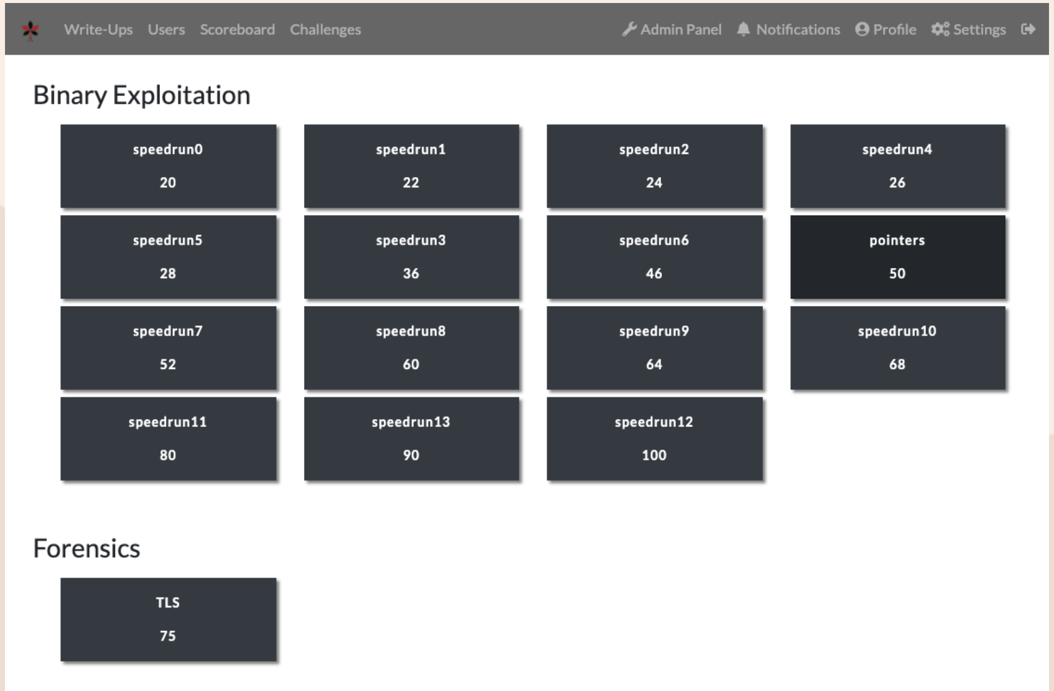
CAPTURE THE FLAG?

Capture the Flag (CTF)



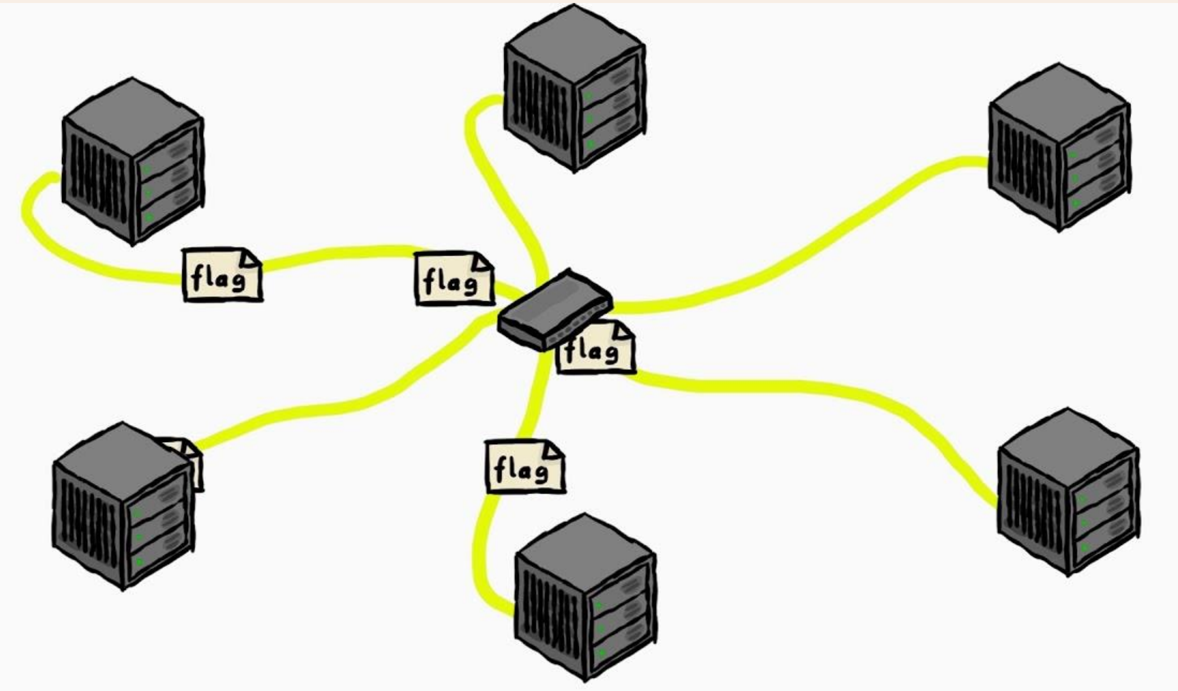
Binary Exploitation

speedrun0 20	speedrun1 22	speedrun2 24	speedrun4 26
speedrun5 28	speedrun3 36	speedrun6 46	pointers 50
speedrun7 52	speedrun8 60	speedrun9 64	speedrun10 68
speedrun11 80	speedrun13 90	speedrun12 100	

Forensics

TLS 75

Jeopardy



Attack-Defense

Jeopardy CTF

Cryptography

Crack the code by decoding or decryption.

Forensics

All about data recovery and analysis.

OSINT

Open Source Intelligence challenges.

Web

Hack a web app via a chain of attacks and exploits.

Reverse Engineering

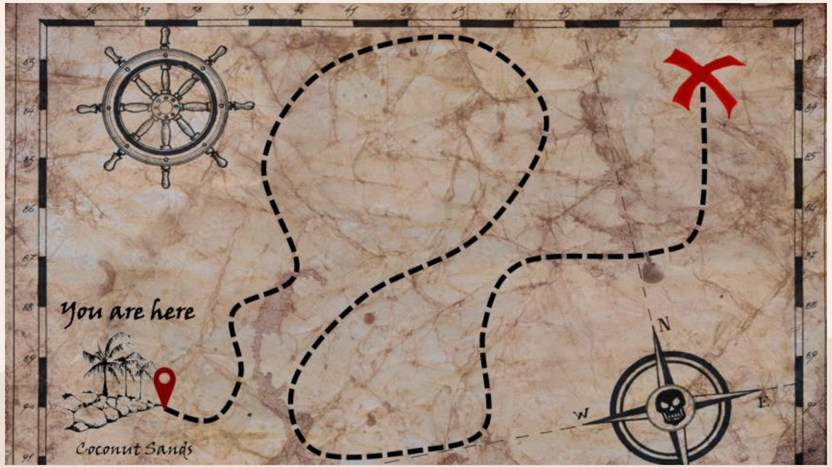
Reverse engineering at its finest.

Binary Exploitation / Pwn

Exploit the program to gain control of a shell.

Your Handy Tools

Whispering Parchment



Treasure Map

You are here
Coconut Sands
N
E
W

OSINT

Open Source Intelligence (OSINT)

- **What is OSINT?**
  - Open-source = Publicly available sources
  - Example: Internet

- **What information do we gather?**
  - People
  - Organization
  - Location
  - etc.

OSINT Weapons

Google

Popular search engine

Yandex

Reverse image search engine

Face recognition search engine

INTERNET ARCHIVE

WayBackMachine

View web archive

Upon reaching Coconut Sands, you notice the Whispering Parchment is glowing and the words on the parchment are updating themselves.

Then, you hear a sound.

Use the Wayback Machine, travel back in time,
To find clues and secrets of the sublime,
And with the search engine of Google's might,
Solve the challenges with all your sight.

And when the tasks are solved and done,
The next location on the map shall come,
So be quick and nimble on thy quest,
For the treasure awaits, be at your best.

Task:
Solve the 3 challenges on OSINT and reveal the next location on the map.

HANDS-ON ACTIVITIES

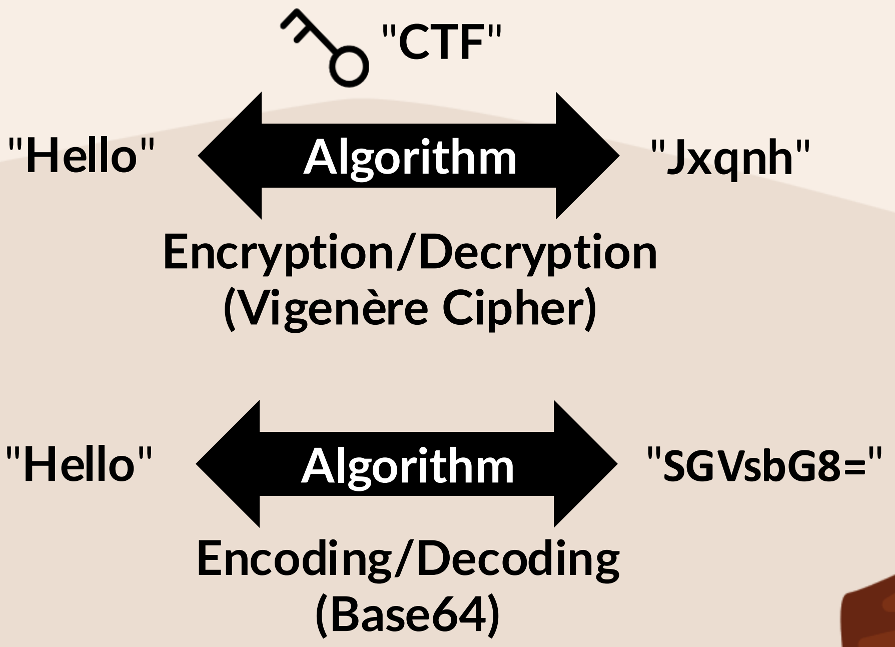
You are here
Temple Ruins
Coconut Sands
N
W
E

CRYPTOGRAPHY

Cryptography

- **What is Cryptography?**
  - Encryption and decryption of data (medium – difficult)
  - Encoding and decoding of data (easy - medium)

- **Common cipher:**
  - Base64
  - Hex (base 16), Binary (base 2)
  - Rot13, Rot47
  - Vigenère cipher


"CTF"

"Hello"

Algorithm

"Jxqnh"

Encryption/Decryption (Vigenère Cipher)

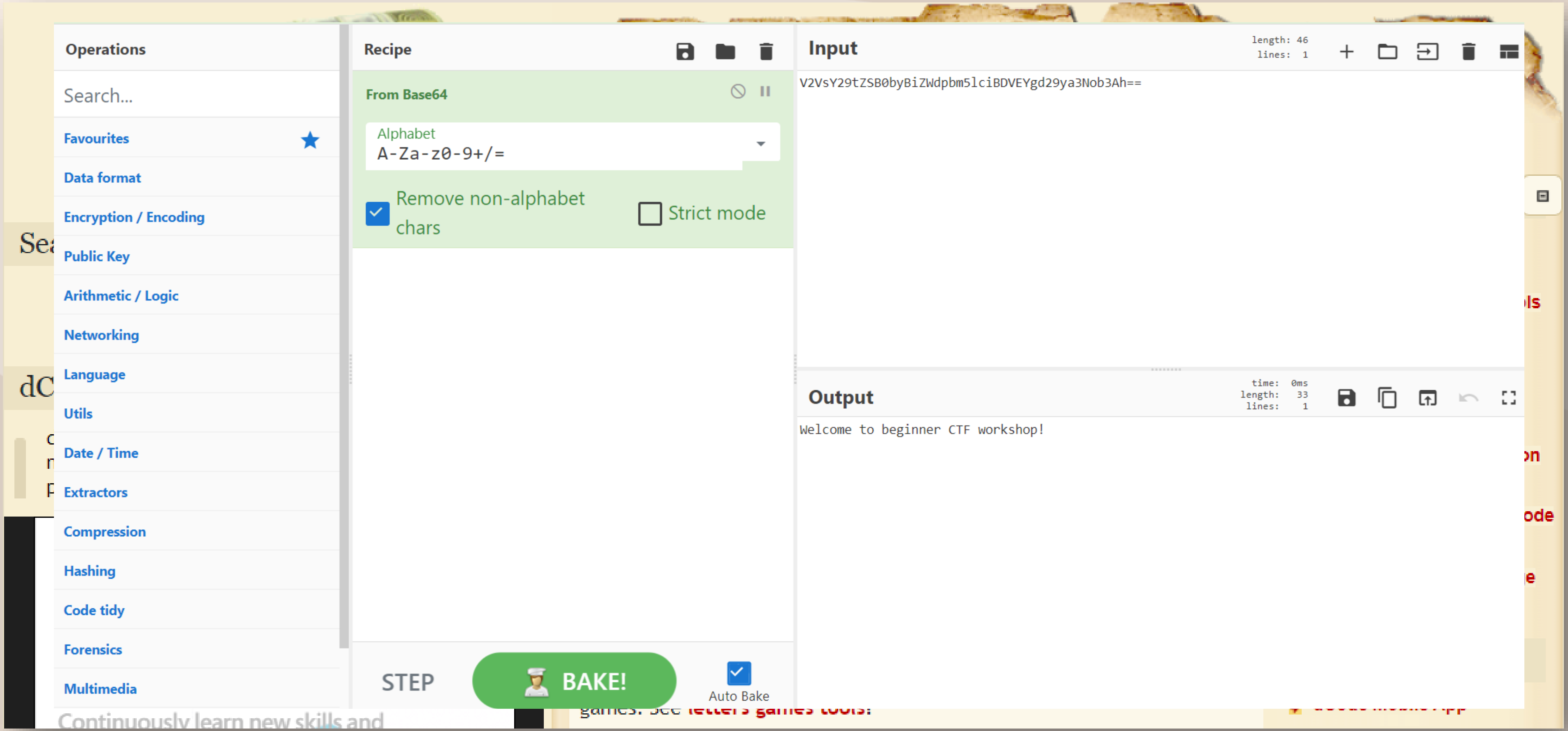
"Hello"

Algorithm

"SGVsbG8="

Encoding/Decoding (Base64)

Cryptography Weapons



From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars ☐ Strict mode

Input (length: 46, lines: 1)

```
V2VsY29tZSB0byBiZWdpbm5lciBDVEYgd29ya3Nob3Ah==
```

Output (time: 0ms, length: 33, lines: 1)

```
Welcome to beginner CTF workshop!
```

CyberChef

Cryptography

Base64

- A way of encoding data
- Represents binary data with 64 characters
- Should only contain A-Z, a-z, 0-9, +, /, =

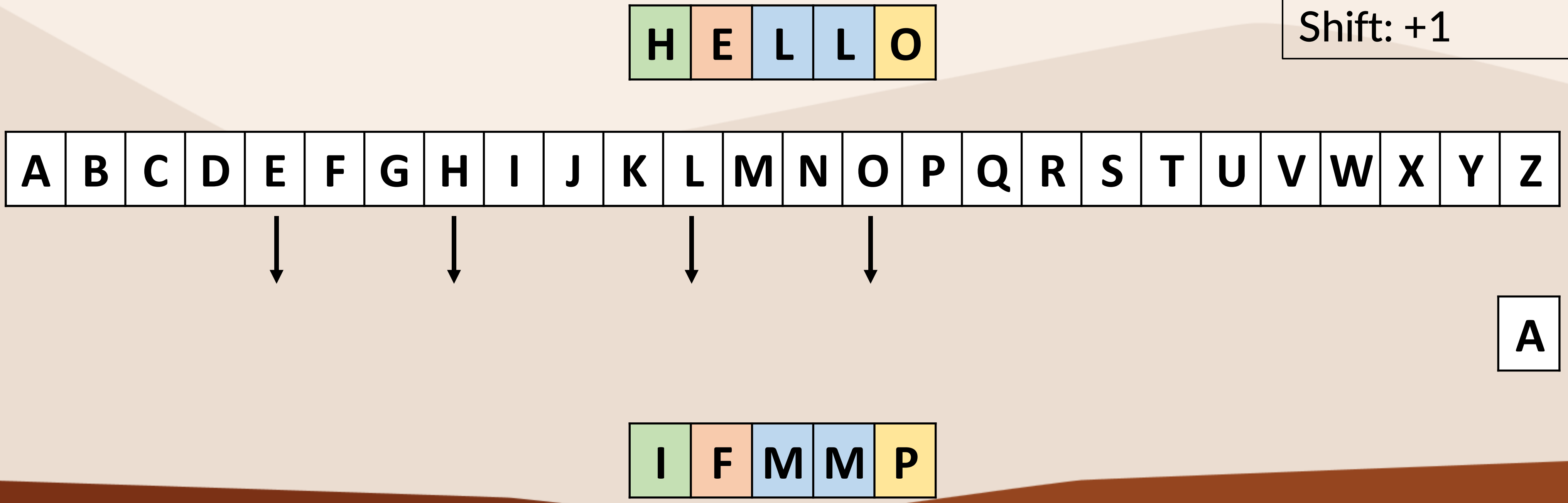
V2VsY29tZSB0byBiZWdpbm5lciBDVEYgd29ya3Nob3Ah==

Cryptography

Caesar Cipher

- Encode by shifting the fixed number of position
- Rot-13: Caesar cipher with a shift of 13

Plain text: HELLO
Shift: +1

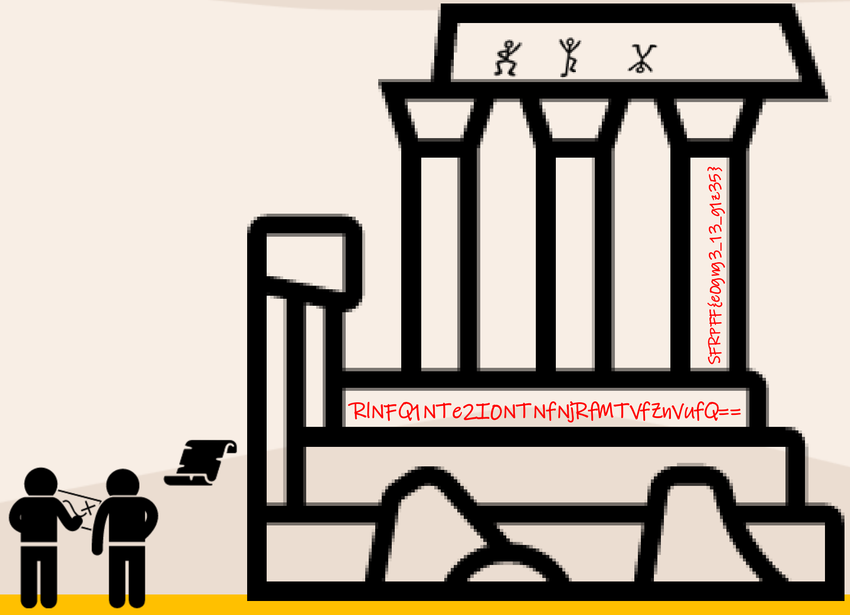


H	E	L	L	O

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

A

I	F	M	M	P



You arrive at Temple Ruins after your short stay in Coconut Sands.

You find symbols, numbers and characters on the walls and pillars of the ruined temple. They seem to have a special meaning…

But fear not, the Whispering Parchment is ready with its hints.

In Temple Ruins, where secrets hide,
Three ciphers must be solved with pride,
Crack the code with skill and might,
And the next location will be in sight.

Task: Decrypt the ciphertexts and reveal the next location on the map.

HANDS-ON ACTIVITIES

Shrouded Sanctuary
You are here
Temple Ruins
Coconut Sands
N
W
E

FORENSICS

Forensics

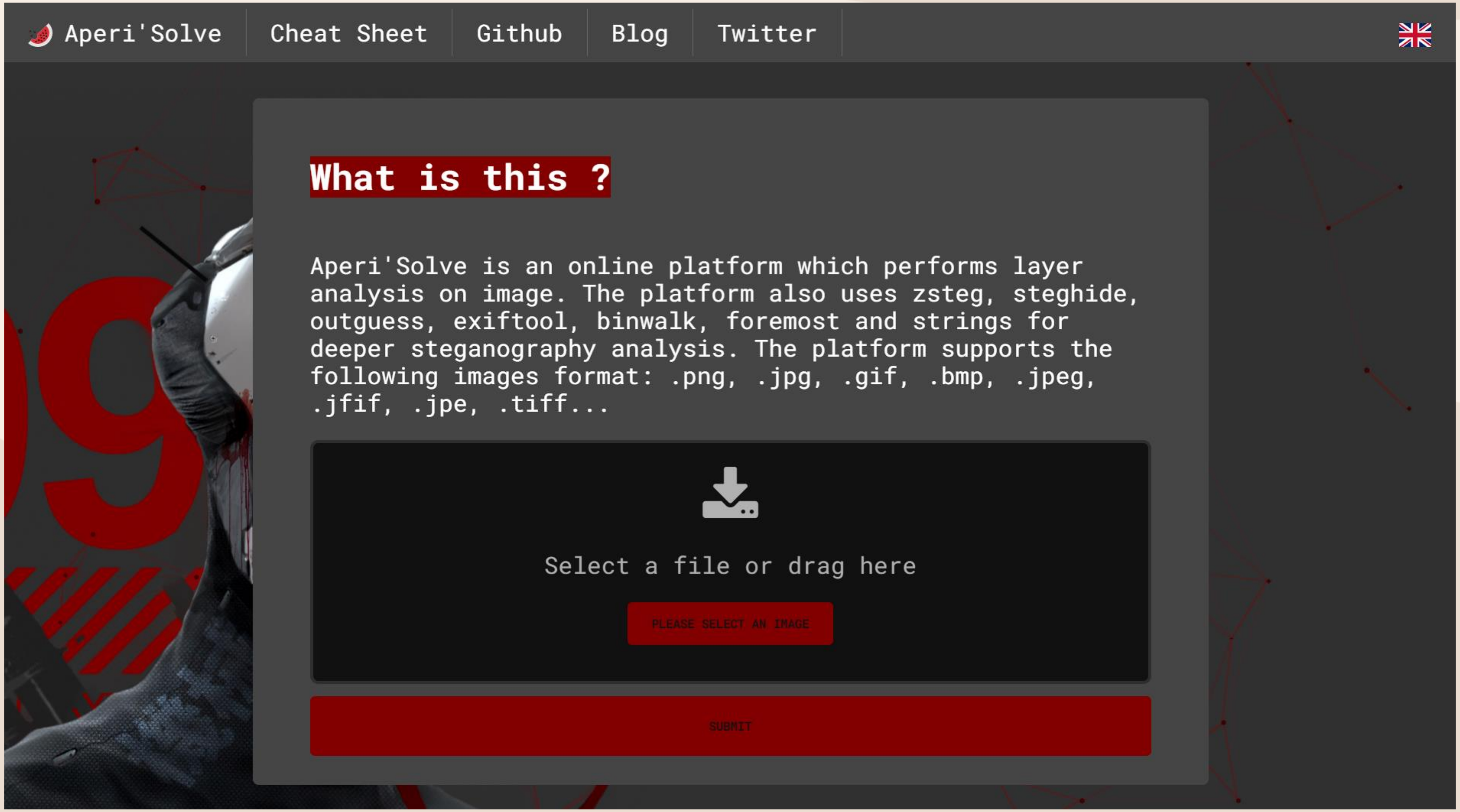
- **What is Forensics?**
  - Analyzing digital artifacts
  - Steganography - finding hidden information in different types of files

- **Common Forensics challenges:**
  - File format analysis
  - Network packet analysis (Wireshark)
  - Memory dump analysis (Volatility)

} intermediate to advanced

* Common tool used

Forensics Weapons



What is this ?

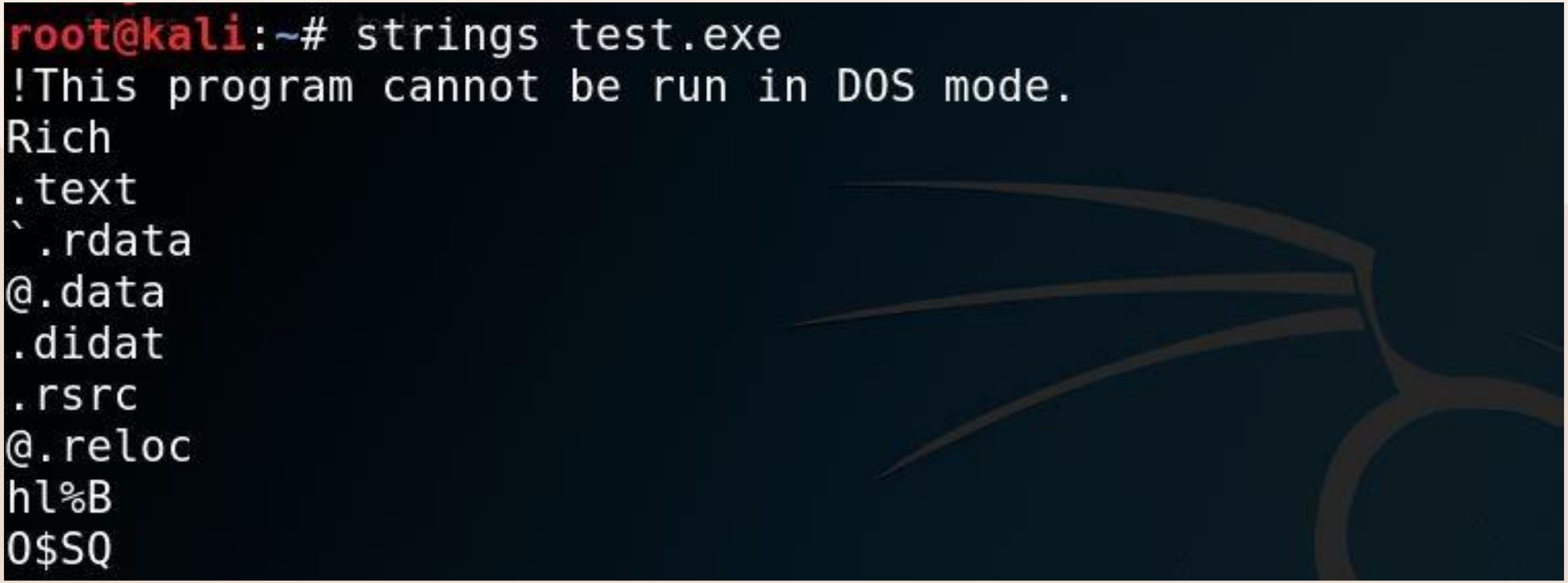
Aperi'Solve is an online platform which performs layer analysis on image. The platform also uses zsteg, steghide, outguess, exiftool, binwalk, foremost and strings for deeper steganography analysis. The platform supports the following images format: .png, .jpg, .gif, .bmp, .jpeg, .jfif, .jpe, .tiff...

AperiSolve

File Format Analysis

Common Things That Should be Done on Basic Challenges:

- Strings

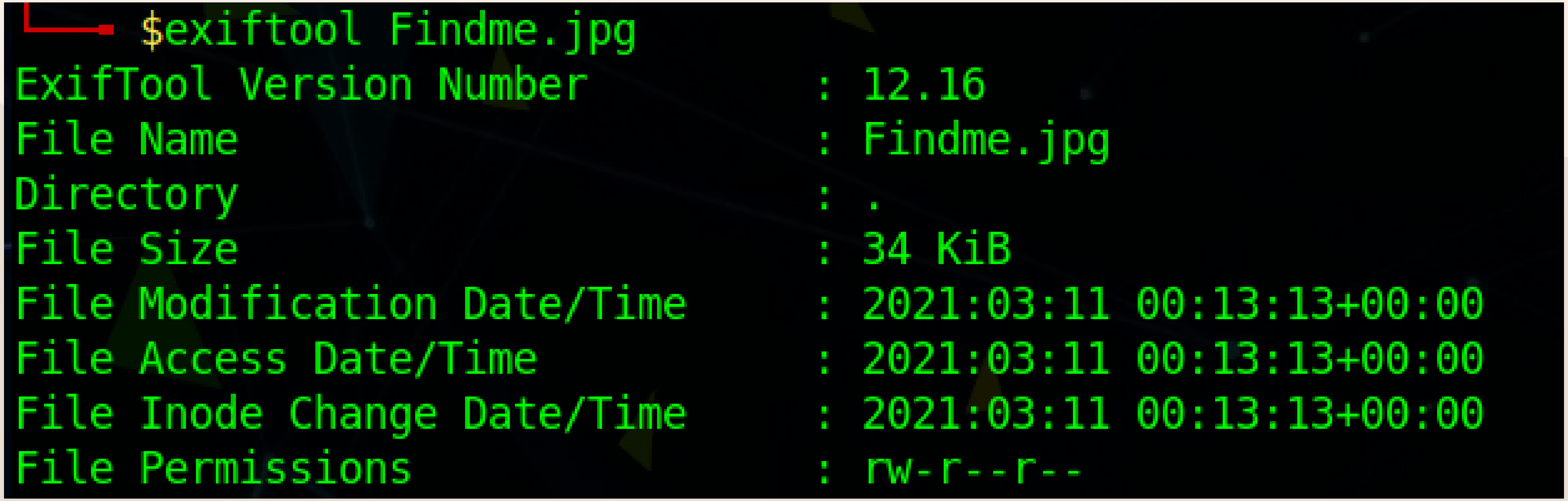


```
root@kali:~# strings test.exe
!This program cannot be run in DOS mode.
Rich
.text
`.rdata
@.data
.didat
.rsrc
@.reloc
hl%B
O$SQ
```

File Format Analysis

Common Things That Should be Done on Basic Challenges:

- Exiftool

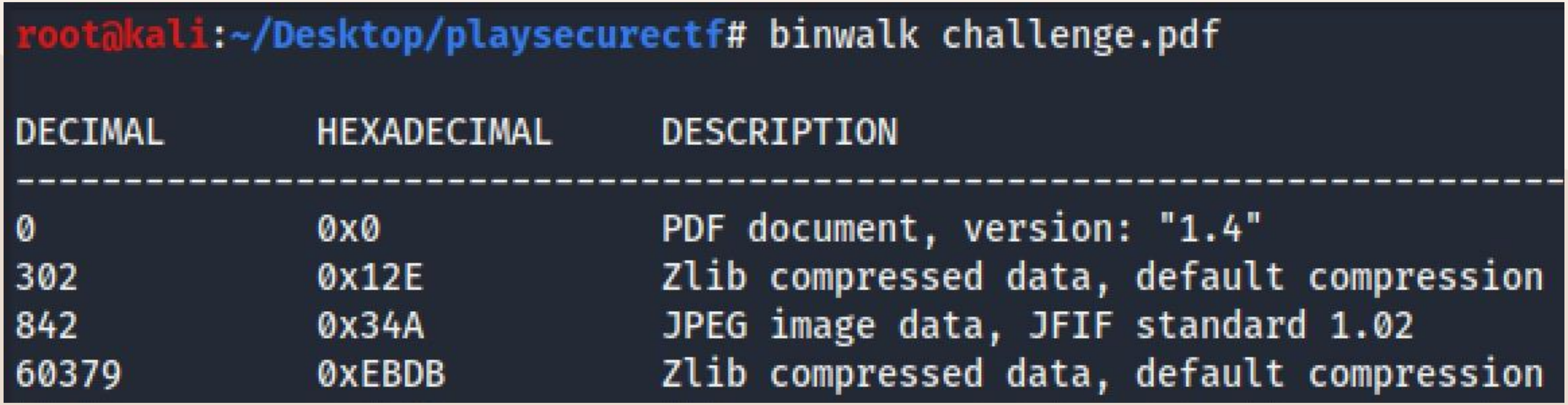


```
└──╼ $exiftool Findme.jpg
ExifTool Version Number         : 12.16
File Name                       : Findme.jpg
Directory                       : .
File Size                       : 34 KiB
File Modification Date/Time     : 2021:03:11 00:13:13+00:00
File Access Date/Time           : 2021:03:11 00:13:13+00:00
File Inode Change Date/Time     : 2021:03:11 00:13:13+00:00
File Permissions                : rw-r--r--
```

File Format Analysis

Common Things That Should be Done on Basic Challenges:

- Binwalk

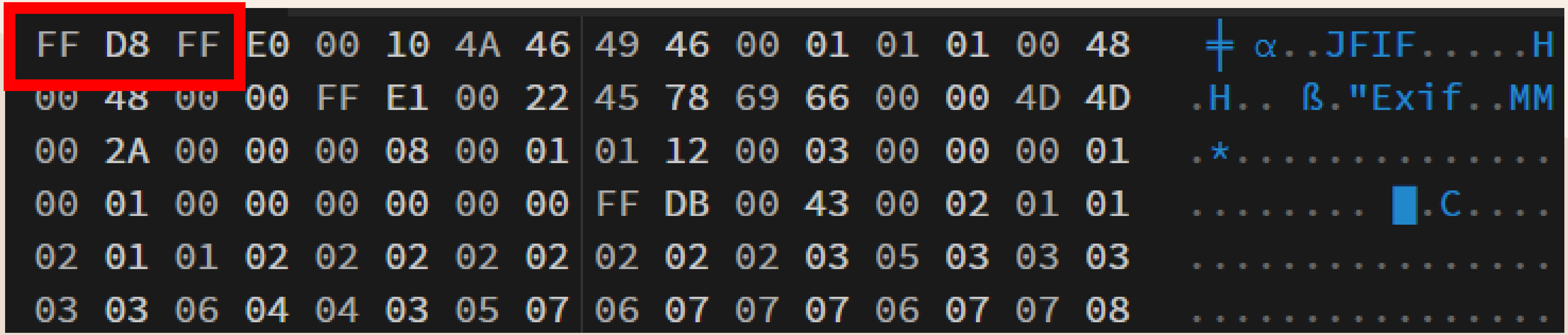


```
root@kali:~/Desktop/playsecurectf# binwalk challenge.pdf

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PDF document, version: "1.4"
302           0x12E           Zlib compressed data, default compression
842           0x34A           JPEG image data, JFIF standard 1.02
60379         0xEBDB          Zlib compressed data, default compression
```

File Format Analysis

Common Things That Should be Done on Basic Challenges:

- Check file header



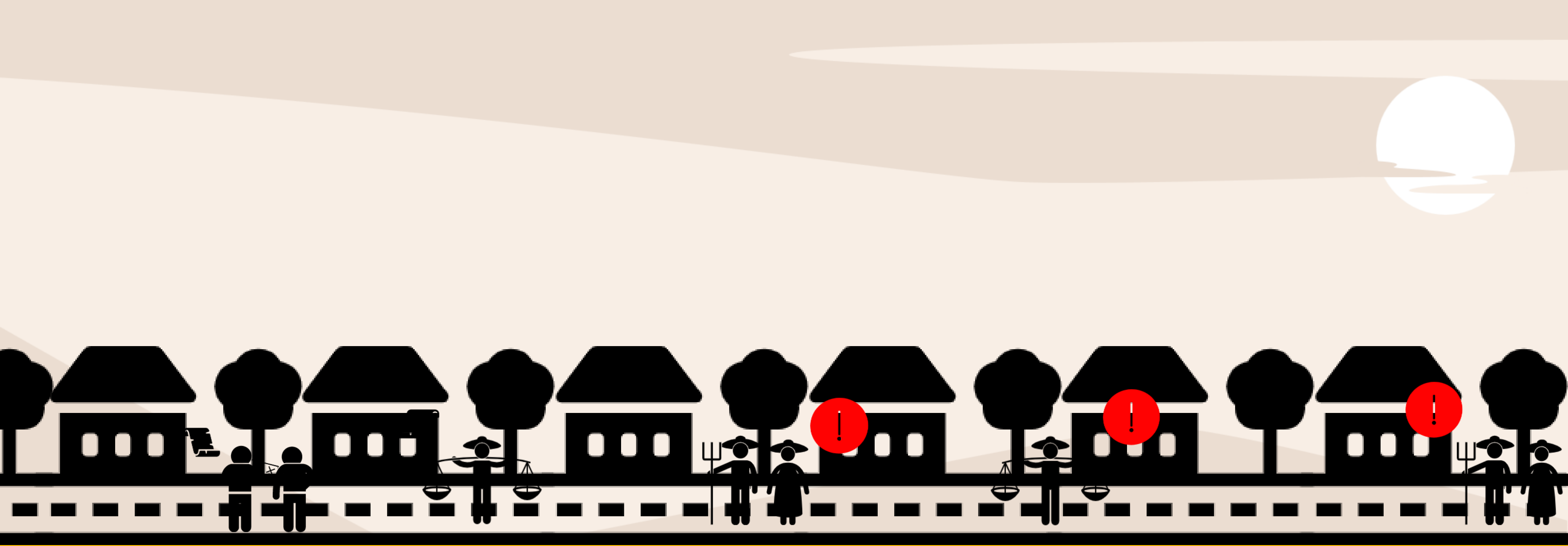
```
FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 00 48    ╪ α..JFIF.....H
00 48 00 00 FF E1 00 22 45 78 69 66 00 00 4D 4D   .H.. ß."Exif..MM
00 2A 00 00 00 08 00 01 01 12 00 03 00 00 00 01   .*..............
00 01 00 00 00 00 00 00 FF DB 00 43 00 02 01 01   ........ █.C....
02 01 01 02 02 02 02 02 02 02 02 03 05 03 03 03   ................
03 03 06 04 04 03 05 07 06 07 07 07 06 07 07 08   ................
```

File Format Analysis

Common Things That Should be Done on Basic Challenges:

- Strings
- Exiftool
- Binwalk
- Check file header

AperiSolve



At Shrouded Sanctuary, you see villagers calling for help.

They are all people with little knowledge on technology and smart devices, much less on digital forensics.

As usual, the Whispering Parchment has the hints you need. Can you help solve their problems?

The first two tests your digital wit,
With **image strings** and **audio** that'll hit.

But the final challenge will test your skill,
Changing file headers with great precision and will.

For it is this that will **reveal the way**,
To the next location, where the treasure may lay.

Task:
Help the villagers with their digital forensic tasks and reveal the next location on the map.

HANDS-ON ACTIVITIES

Shrouded Sanctuary
Temple Ruins
Coconut Sands
Sentinel Beacon
You are here
N
W
E

WEB EXPLOITATION

Web Exploitation

What is Web Exploitation?

The practice of finding and exploiting vulnerabilities in web applications or websites.

Common Web Exploitation challenges:

- Programming Malpractices
- SQL Injection (SQLi)
- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF)

} Advanced

After leaving Shrouded Sanctuary, you move on to the next location on the map, Sentinel Beacon, a lighthouse near the southward shores of the island.

You notice that the door to the lighthouse is locked, barring your entry. Suddenly, a sound came from the lighthouse and it seems to be asking for a password. You tried guessing the password but you failed miserably. Then, you notice something written on the walls.

The lighthouse is ALIVE

To: My Future Self

If you have forgotten the password to enter the lighthouse, check out the lighthouse website.

It seems that the lighthouse guard has left a note to remind himself/herself of the lighthouse password in case that he/she has forgotten about it.

Next, you check the Whispering Parchment for the tasks at this lighthouse.

Onward, to the web's domain,
Where secrets hide in code arcane.

The first two tests the Sentinel Beacon will unseal,
And lead you closer to the map's next revealed detail.

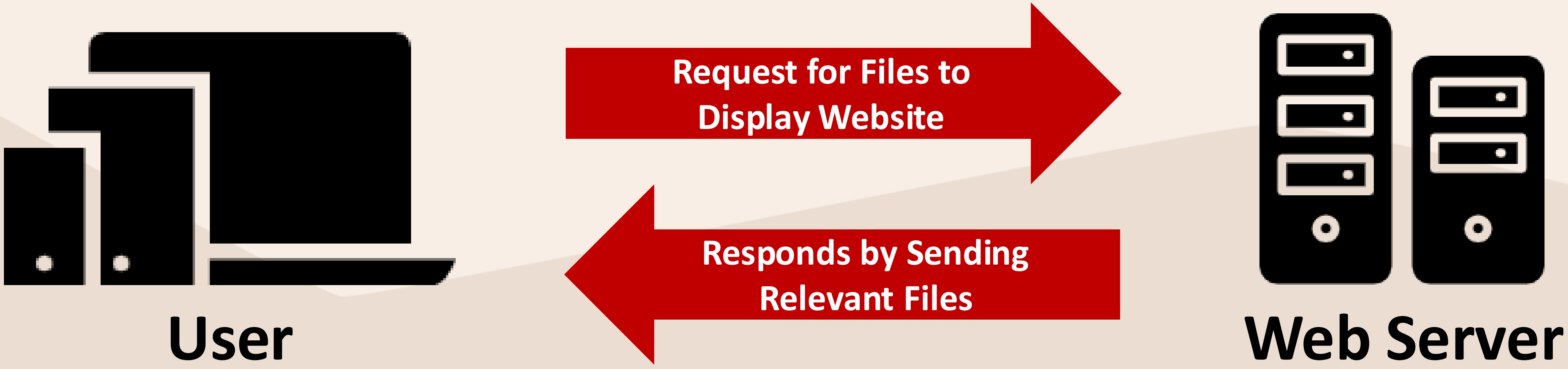
The last and final trial will test your SQL might,
For only those who know its ways can see the light.

Task:
Find the two passwords on Sentinel Beacon's website and solve the SQL challenge to reveal the next location on the map.

HANDS-ON ACTIVITIES

How is a Website Displayed?

1. The user enters a website URL/clicks on a link. Then, the web browser sends request to the server hosting the website for the website files.



Request for Files to Display Website

Responds by Sending Relevant Files

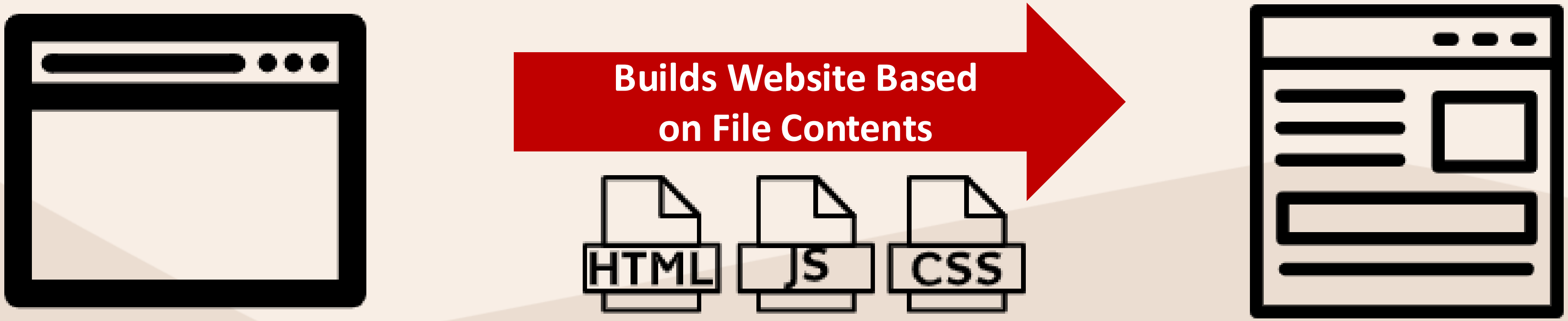
User

Web Server

2. The server responds with the relevant files of the requested website.

How is a Website Displayed?

3. Using the files sent by the server, the web browser structures and designs the website accordingly.



Web Browser

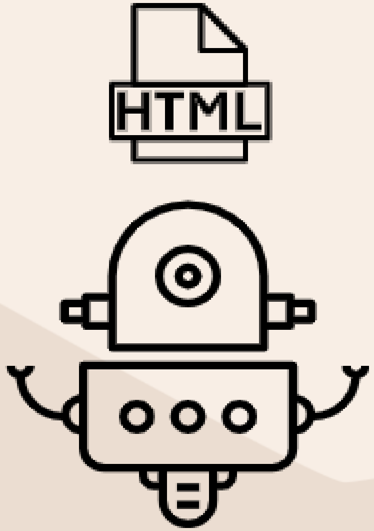
Website Generated

4. The complete website as designed and structured by its developer is displayed in the user's web browser.

Imagine that Webpages are Robots

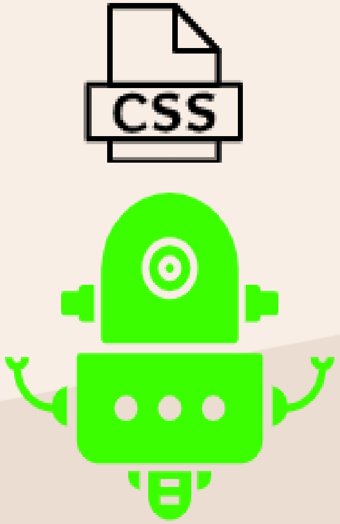
To create a robot, you need to:

<Architect>



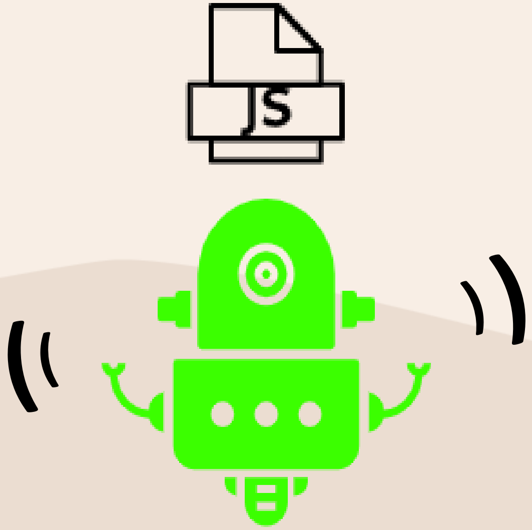
Step 1:
Determine
the features of
the robot

<Stylist>



Step 2:
Determine
the design of
the robot

<Engineer>



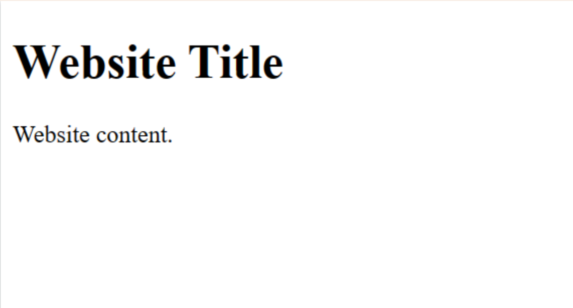
Step 3:
Determine how the
robot moves and
interacts with humans

Imagine that Webpages are Robots

Same concept for webpages:

<Architect>

HTML



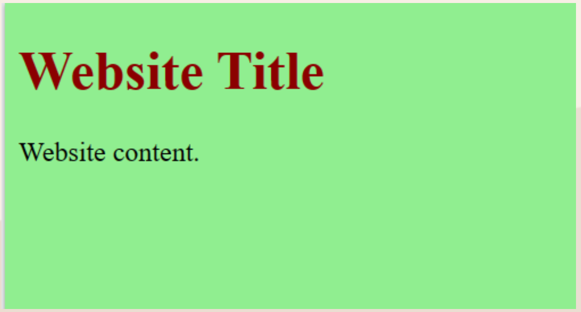
Website Title

Website content.

Step 1:
Determine
the features of the
webpage

<Stylist>

CSS



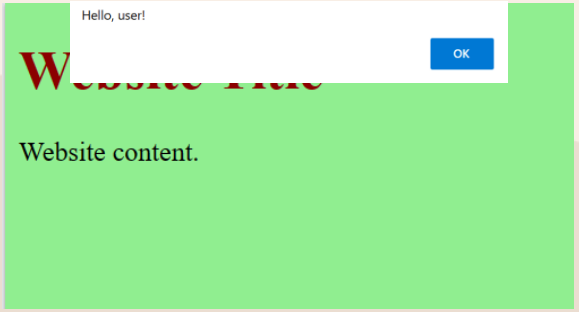
Website Title

Website content.

Step 2:
Determine
the design of
the webpage

<Engineer>

JS



Hello, user!

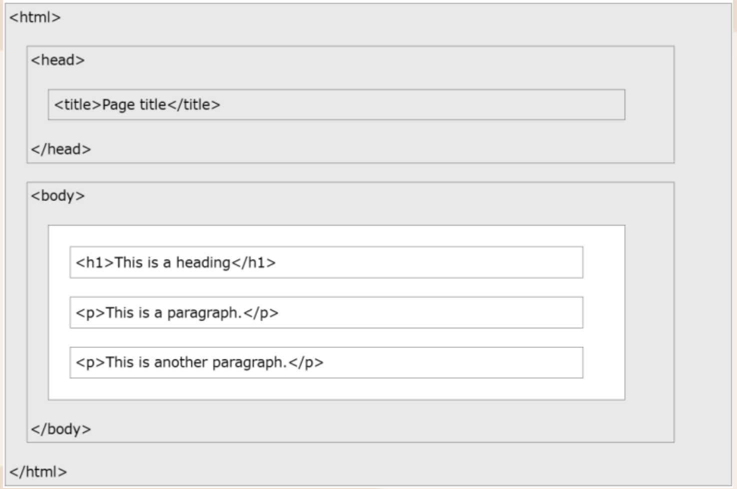
Website content.

Step 3:
Determine how the
webpage interacts with
humans

Website Development

Hypertext Markup Language (HTML)

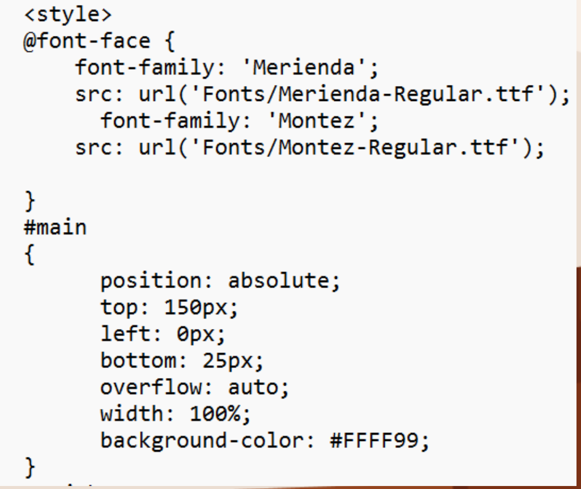
- Defines the structure of the webpage



```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

Cascading Style Sheets (CSS)

- Defines the visual appearance of the webpage



```
<style>
@font-face {
    font-family: 'Merienda';
    src: url('Fonts/Merienda-Regular.ttf');
      font-family: 'Montez';
    src: url('Fonts/Montez-Regular.ttf');

}
#main
{
      position: absolute;
      top: 150px;
      left: 0px;
      bottom: 25px;
      overflow: auto;
      width: 100%;
      background-color: #FFFF99;
}
```

JavaScript (JS)

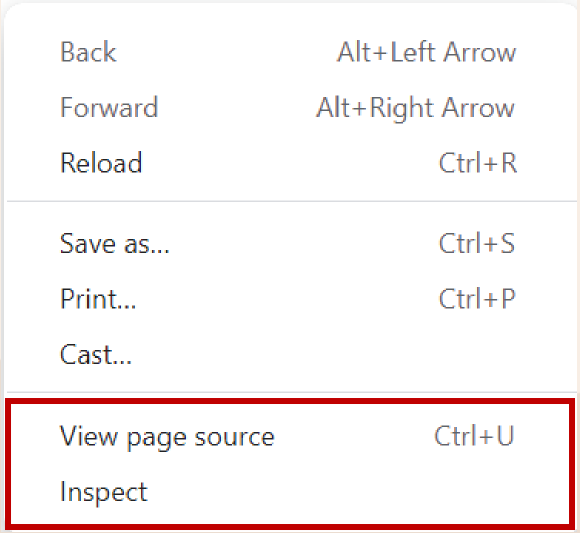
- Provides interactivity and dynamic functionality

This page says

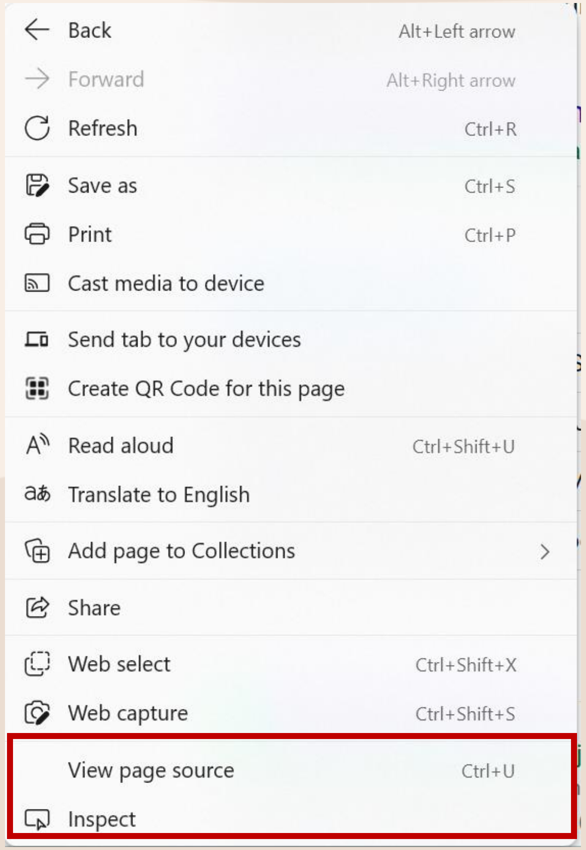
Wrong Password

These are often called the building blocks of the Web.

Viewing Source Code

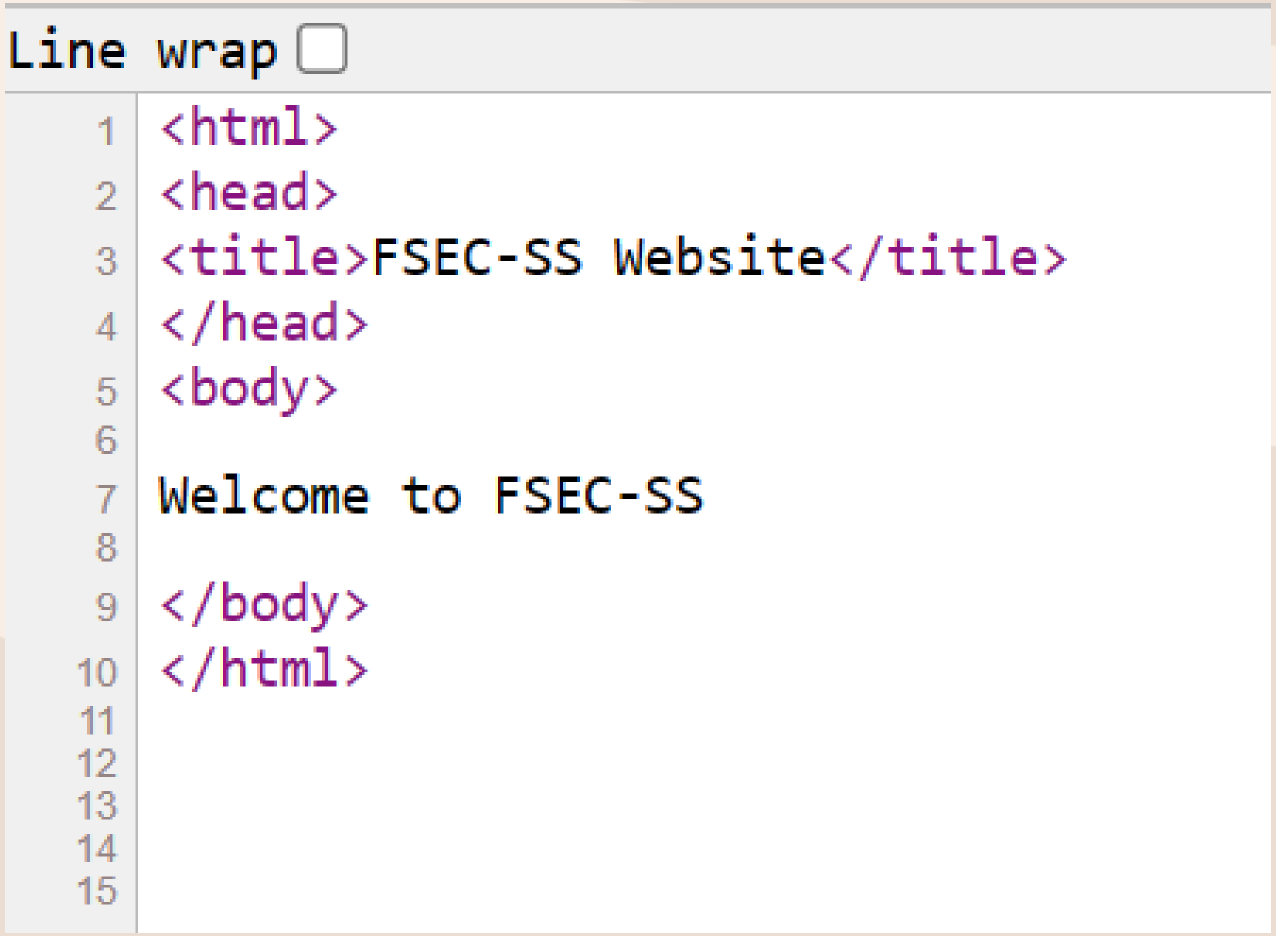


Google Chrome



Microsoft Edge

Viewing Source Code (View Page Source)

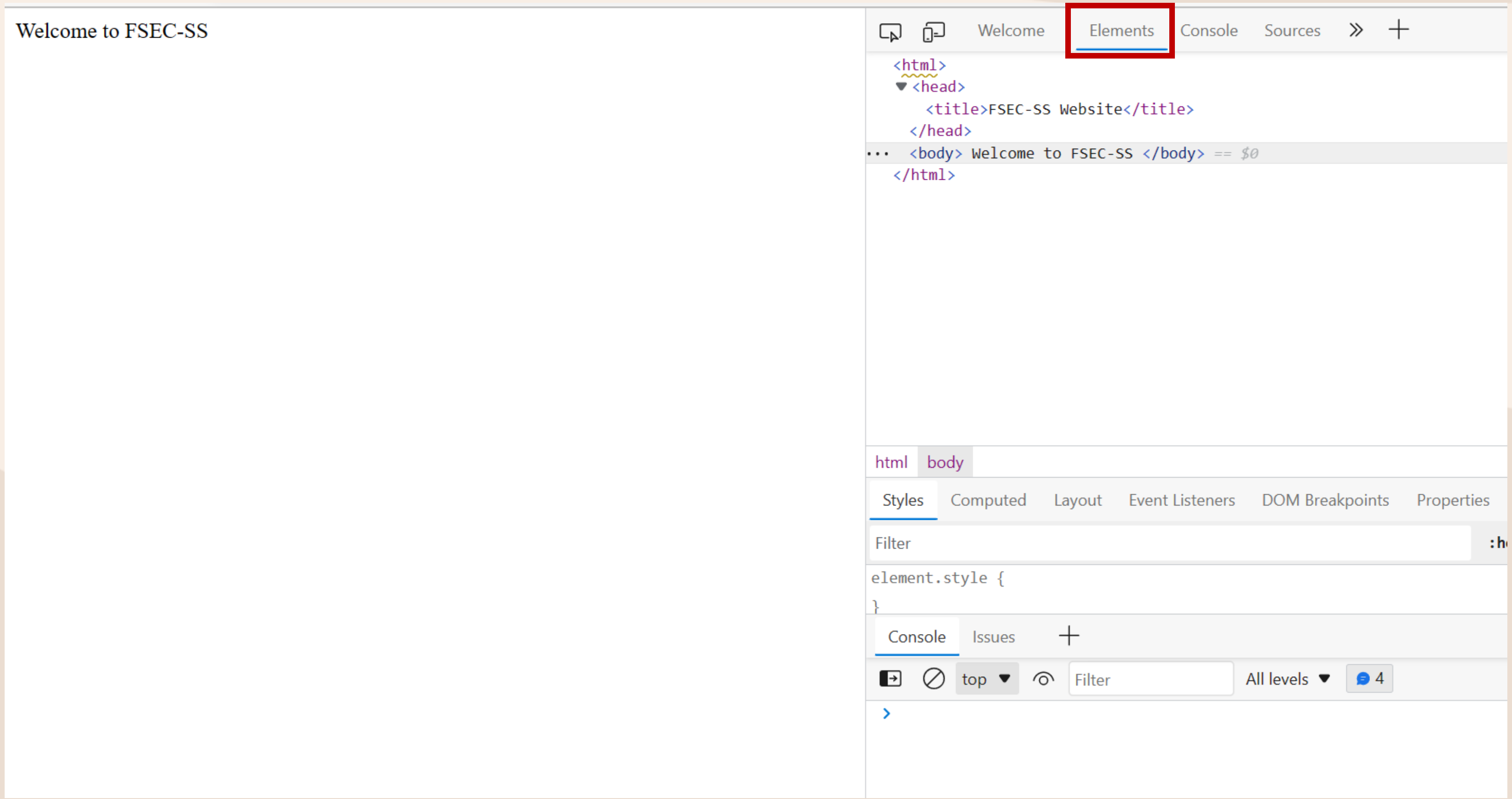


Line wrap ☐

```
<html>
<head>
<title>FSEC-SS Website</title>
</head>
<body>

Welcome to FSEC-SS

</body>
</html>
```

Inspect Page Source

Viewing Source Code (Inspect - Elements)

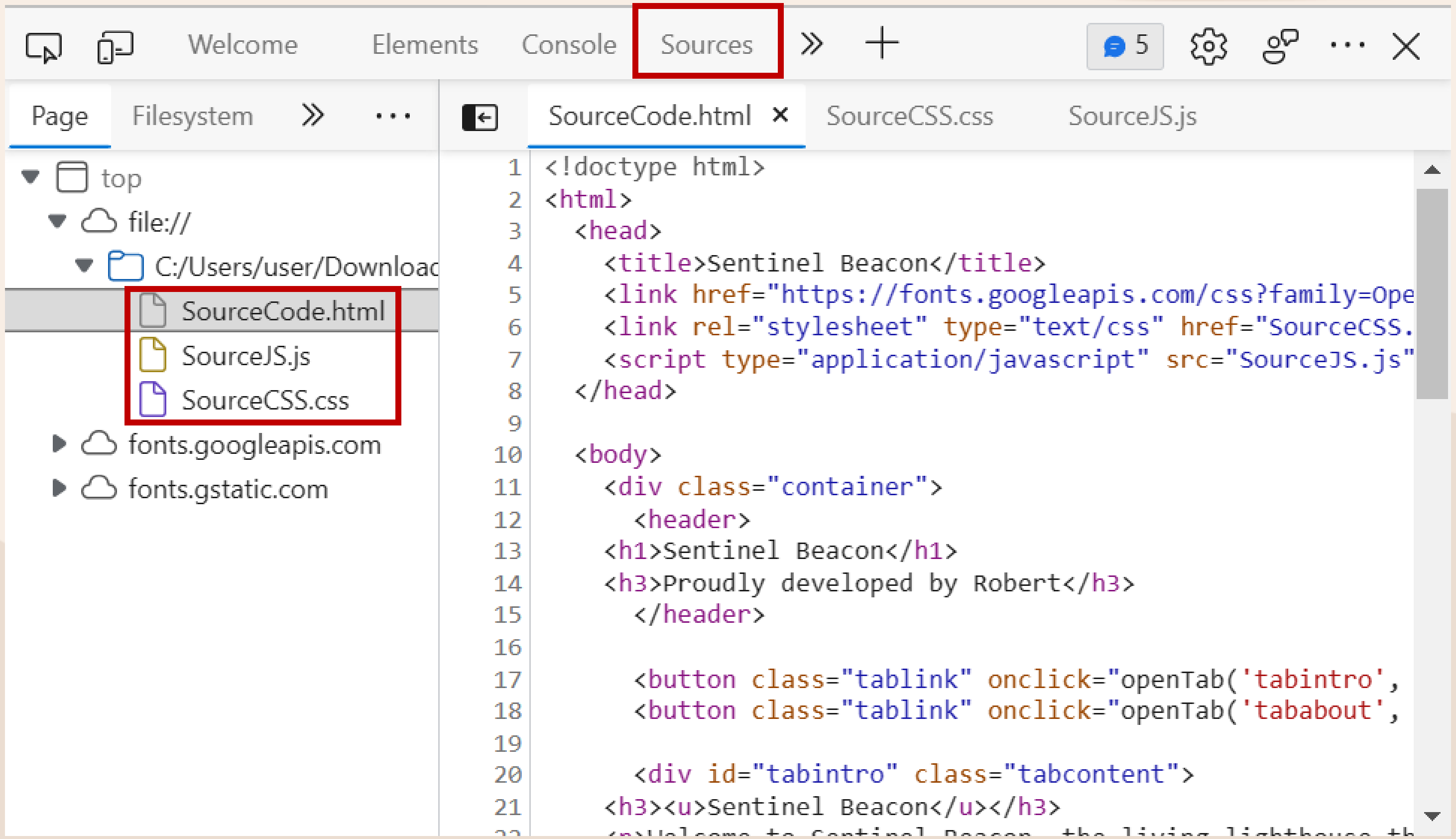

Welcome to FSEC-SS

```
<html>
 ▼<head>
    <title>FSEC-SS Website</title>
   </head>
...  <body> Welcome to FSEC-SS </body> == $0
</html>
```

```
element.style {
}
```

Inspect (Elements)

Viewing Source Code (Inspect - Sources)



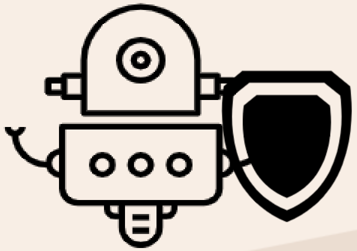
```
<!doctype html>
<html>
  <head>
    <title>Sentinel Beacon</title>
    <link href="https://fonts.googleapis.com/css?family=Ope
    <link rel="stylesheet" type="text/css" href="SourceCSS.
    <script type="application/javascript" src="SourceJS.js"
  </head>

  <body>
    <div class="container">
      <header>
    <h1>Sentinel Beacon</h1>
    <h3>Proudly developed by Robert</h3>
      </header>

      <button class="tablink" onclick="openTab('tabintro',
      <button class="tablink" onclick="openTab('tababout',

      <div id="tabintro" class="tabcontent">
    <h3><u>Sentinel Beacon</u></h3>
    <p>Welcome to Sentinel Beacon, the living lighthouse th
```

Inspect (Sources)

robots.txt

- robots.txt is a text file to tell robots (web crawlers and search engine bots) which pages on a website they are allowed to look at (access and index).
  ***Web crawlers: Programs collecting data for search engines**

Website's Guard Robot



Search Engine's Spy Robot

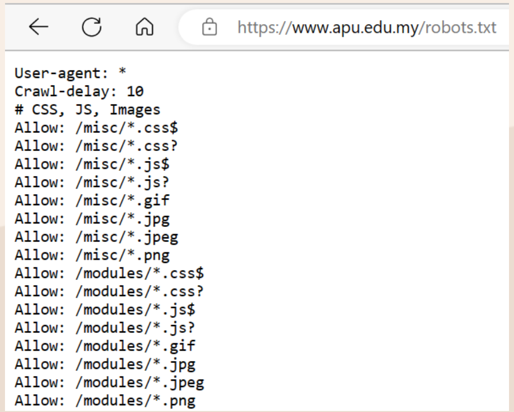
- Think of it like a map for robots to know where they can go and where they cannot go on a website.
- It comprises of two main parts:
  - User-agent (the name of the search engine bot)
  - Disallow directive (URL paths not allowed to access)

robots.txt

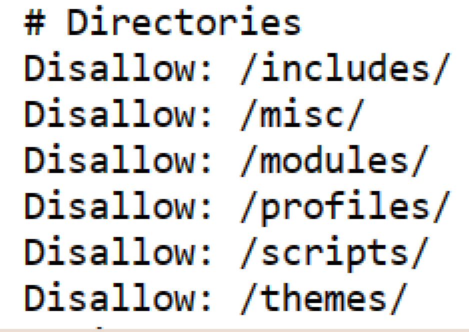
For example:

If a website owner does not want Google to crawl their login page, they can add the following instruction to their robots.txt file:

User-agent: Googlebot
Disallow: /login



```
User-agent: *
Crawl-delay: 10
# CSS, JS, Images
Allow: /misc/*.css$
Allow: /misc/*.css?
Allow: /misc/*.js$
Allow: /misc/*.js?
Allow: /misc/*.gif
Allow: /misc/*.jpg
Allow: /misc/*.jpeg
Allow: /misc/*.png
Allow: /modules/*.css$
Allow: /modules/*.css?
Allow: /modules/*.js$
Allow: /modules/*.js?
Allow: /modules/*.gif
Allow: /modules/*.jpg
Allow: /modules/*.jpeg
Allow: /modules/*.png
```



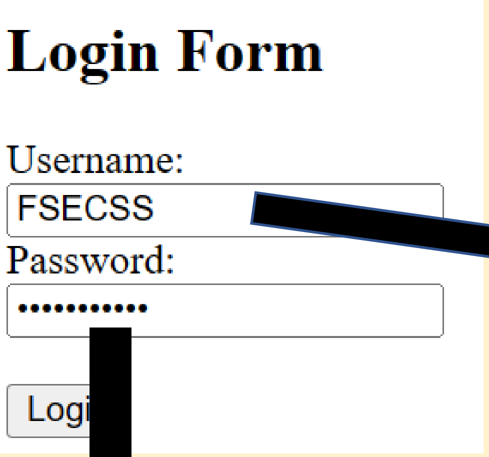
```
# Directories
Disallow: /includes/
Disallow: /misc/
Disallow: /modules/
Disallow: /profiles/
Disallow: /scripts/
Disallow: /themes/
```

Anyone can look at this file so...

NEVER PUT CONFIDENTIAL INFORMATION IN THIS FILE!!!

Login Form

1



Login Form

Username:

FSECSS

Password:

•••••••••••

User enters their username and password in a login form and submits the form to login a system.

2

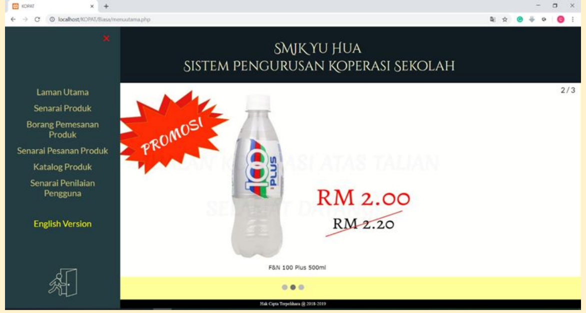
Program (check_login.php)

```
SELECT * FROM
users WHERE username =
'$form_uname' AND
password = '$form_pwd'
```

SQL query is used to retrieve data from the users table where the username and password matches with the user input from the form.

3

if(mysqli_num_rows($result) > 0)



SMJK YU HUA
SISTEM PENGURUSAN KOPERASI SEKOLAH

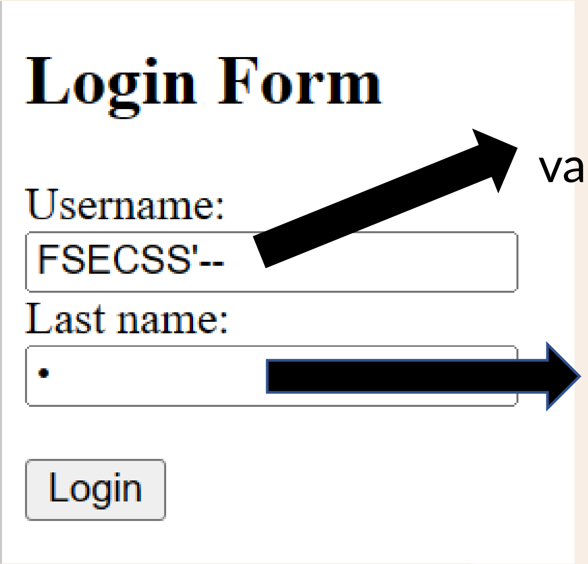
RM 2.00
~~RM 2.20~~

F&N 100 Plus 500ml

Hak Cipta Terpelihara @ 2018-2019

If there is such a record that matches the user input in the database, lead the user to the homepage. Otherwise, the user login attempt will fail.

SQL Injection (SQLi)



Login Form

Username: FSECSS'--

Last name: •

Stored in the variable "form_uname"

Stored in the variable "form_pwd"

Program (check_login.php)

```
SELECT * FROM users WHERE
username = '$form_uname' AND
password = '$form_pwd'
```

```
SELECT * FROM users WHERE username
= 'FSECSS' -- AND password = '-'
```

The password does not matter anymore because the SQL query just retrieves all records where the username is 'admin'

"--" indicates the start of an SQL comment so whatever that comes after "--" will be ignored by the compiler

username	password
FSECSS	ilovefsecss

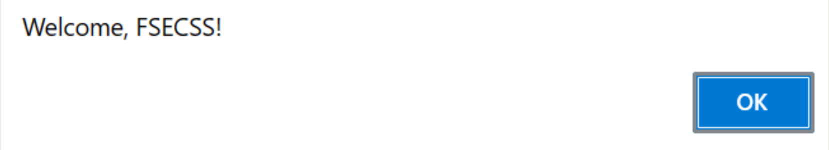
SQL Injection (SQLi)

Program (check_login.php)

```
if(mysqli_num_rows($result) > 0)
{
   echo '<script> alert("Welcome,
   '.$form_uname.'!!");
   window.location.href="homepage.php";
   </script>';
}
```

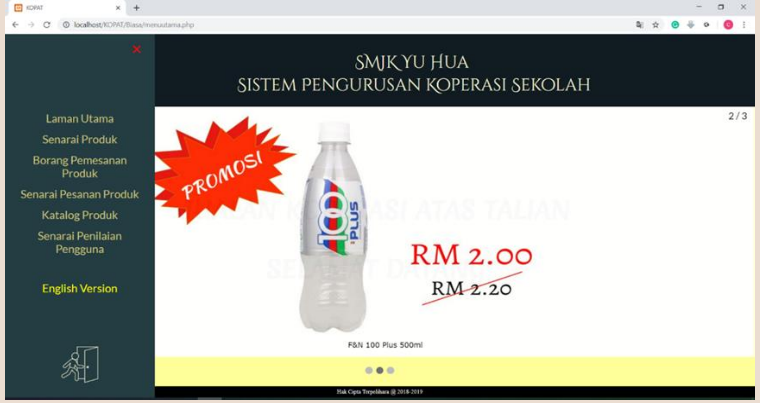
If **any rows/records** are found to **match the username (FSECSS)**, display a welcome message and bring the user to the website homepage

Result:

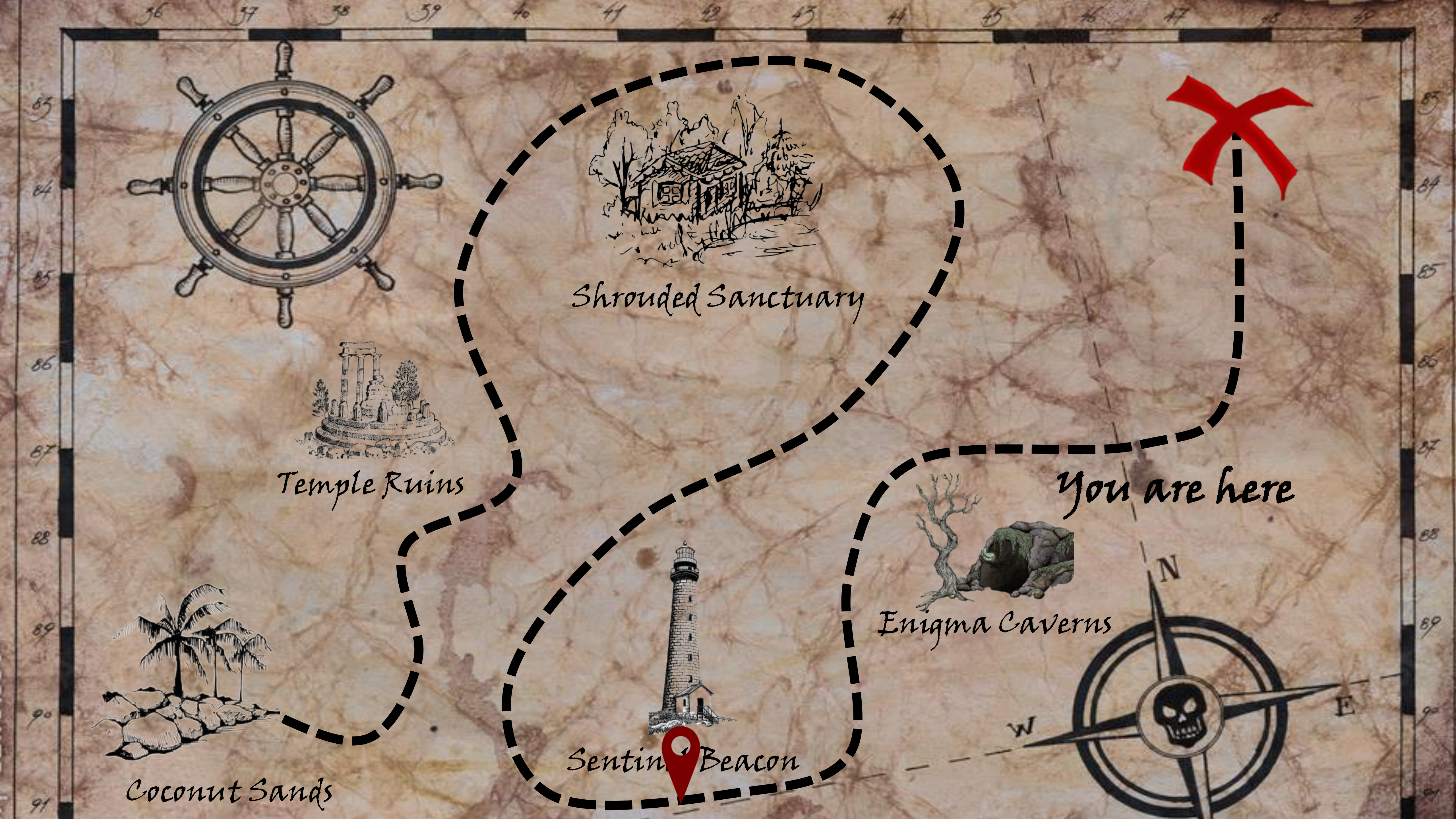


Welcome, FSECSS!

Welcome Message Popup Alert

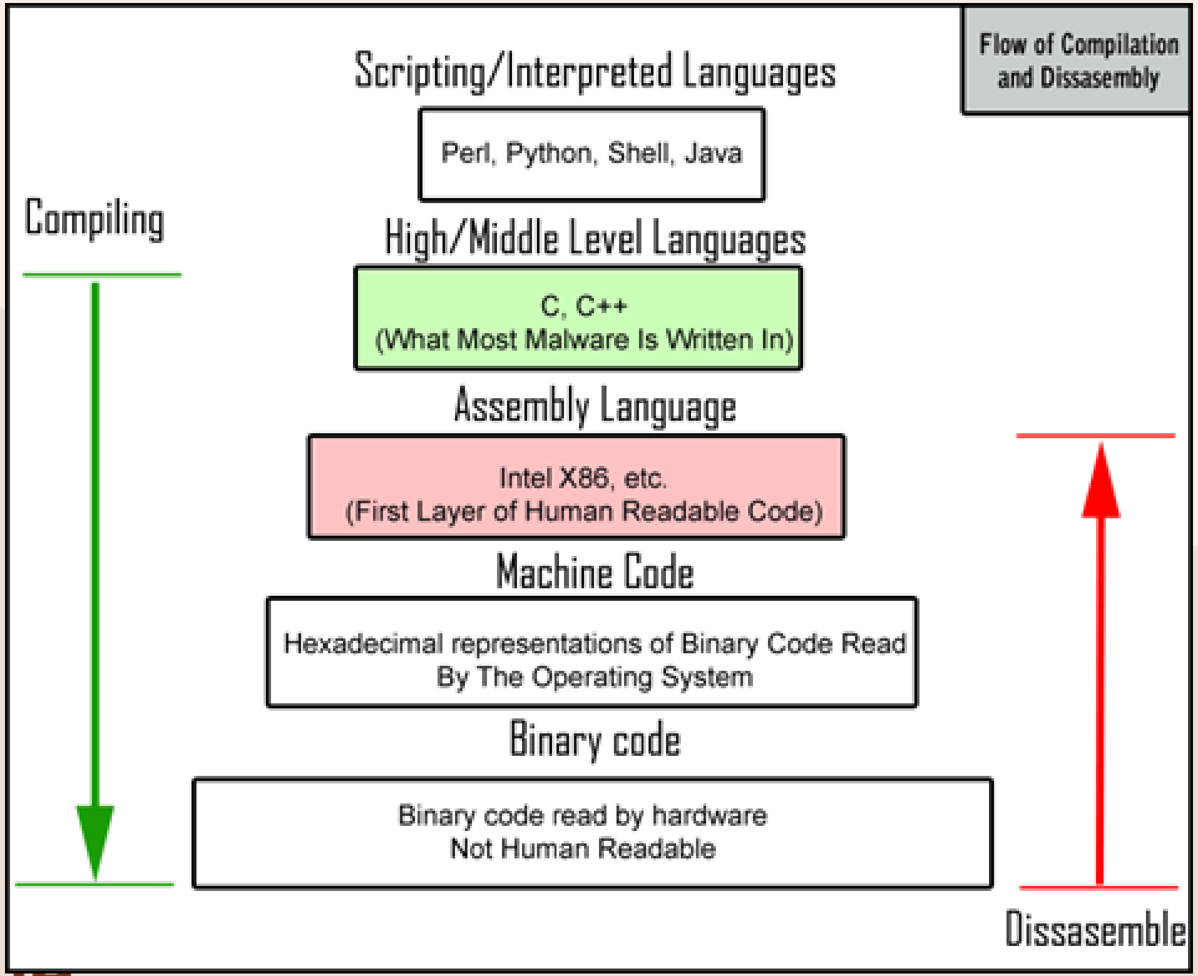


Sample Website Homepage


Shrouded Sanctuary
Temple Ruins
You are here
Enigma Caverns
Coconut Sands
Sentinel Beacon
N
W
E

REVERSE ENGINEERING

Reverse Engineering (RE)

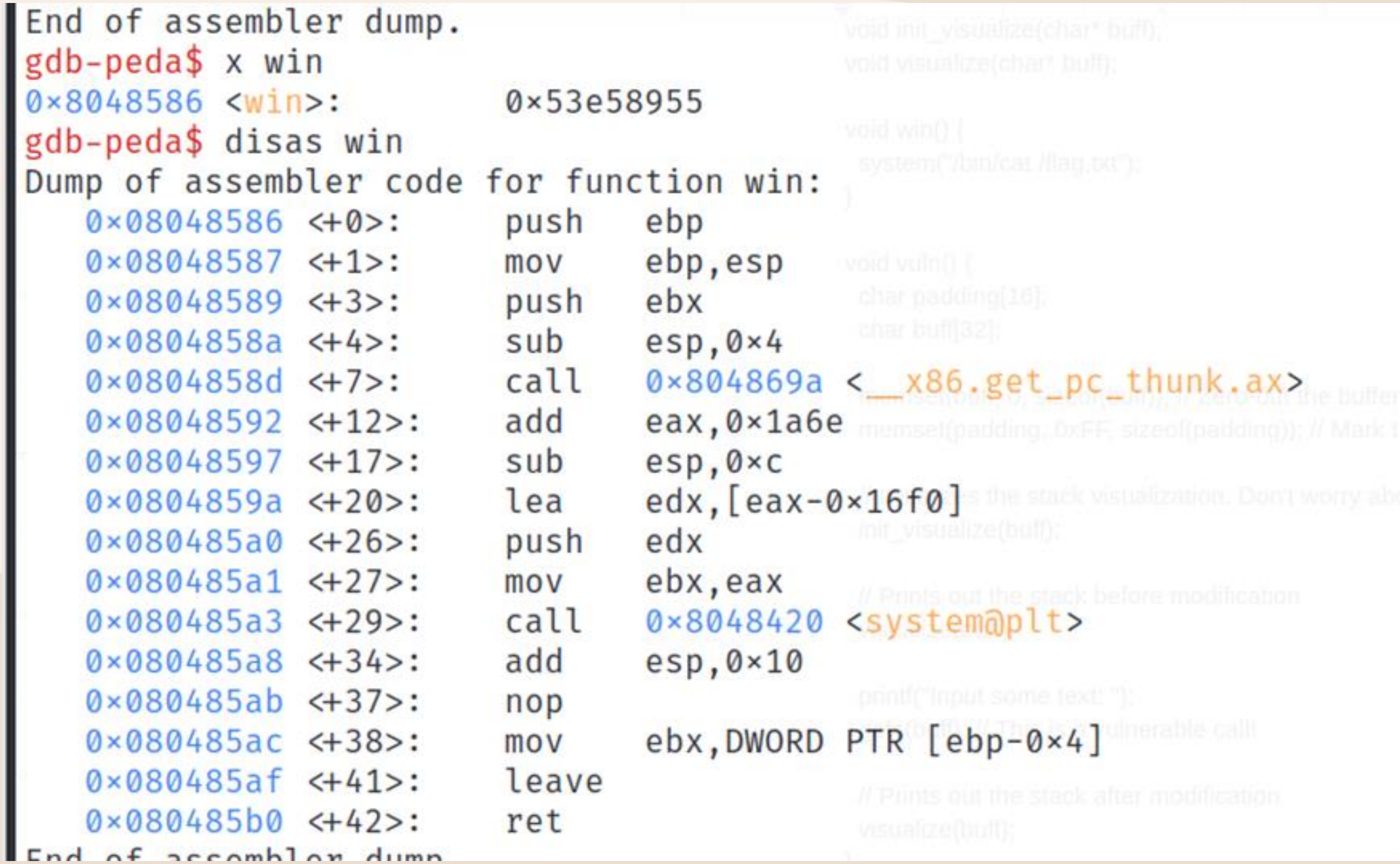


Flow of Compilation and Dissasembly
Scripting/Interpreted Languages
Perl, Python, Shell, Java
High/Middle Level Languages
C, C++
(What Most Malware Is Written In)
Assembly Language
Intel X86, etc.
(First Layer of Human Readable Code)
Machine Code
Hexadecimal representations of Binary Code Read By The Operating System
Binary code
Binary code read by hardware
Not Human Readable
Compiling
Dissasemble

CAUTION: DIFFICULT CTF CATEGORY

What is Reverse Engineering in CTF?
Given executable files (.exe/.ELF), players analyse low-level binary code/assembly code to understand how it works without knowing the high-level source code. Normally, there will be some hidden information leading players to the final flag.

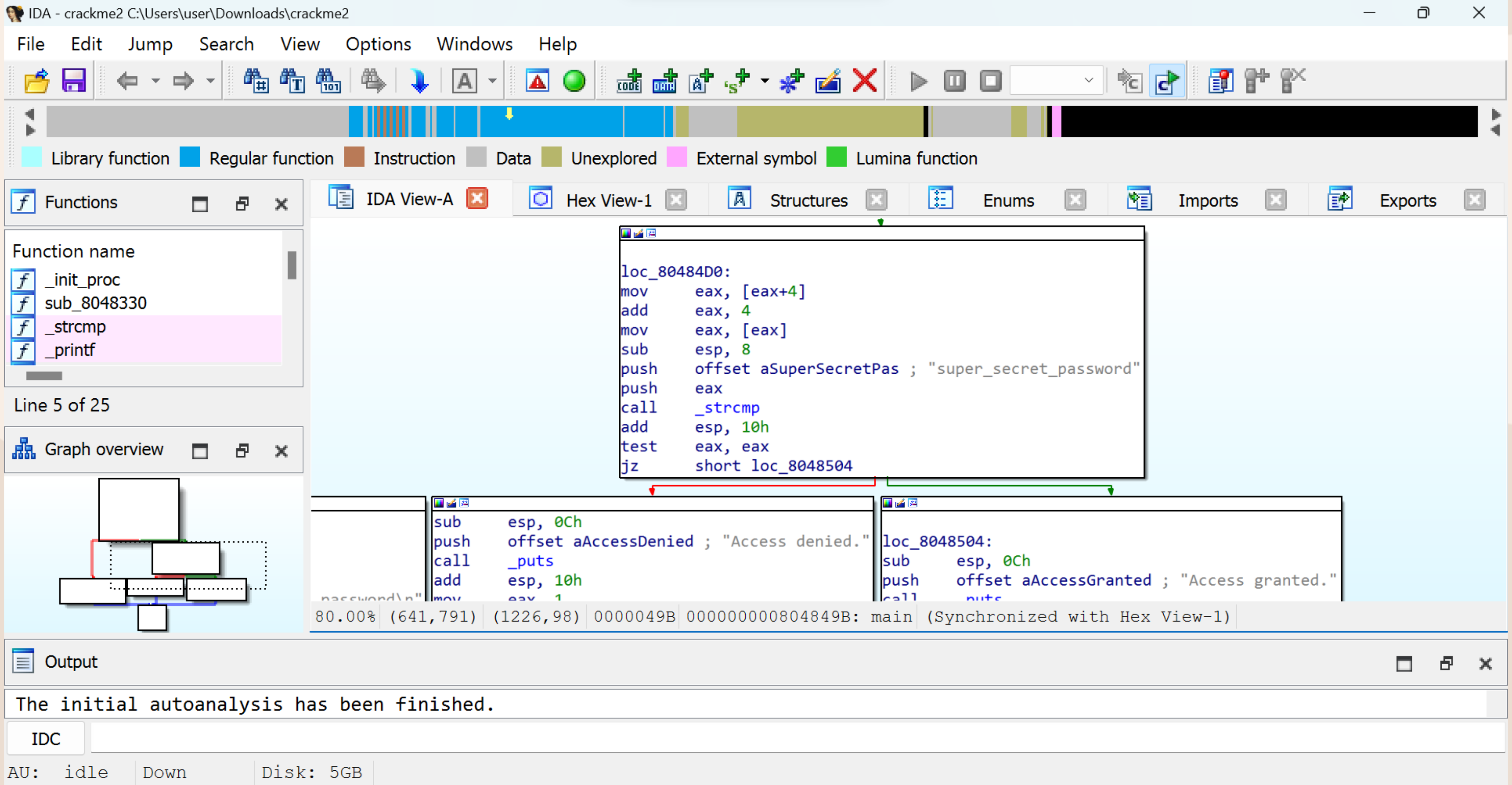
Reverse Engineering Weapons



```
End of assembler dump.
gdb-peda$ x win
0x8048586 <win>:        0x53e58955
gdb-peda$ disas win
Dump of assembler code for function win:
   0x08048586 <+0>:     push   ebp
   0x08048587 <+1>:     mov    ebp,esp
   0x08048589 <+3>:     push   ebx
   0x0804858a <+4>:     sub    esp,0x4
   0x0804858d <+7>:     call   0x804869a <__x86.get_pc_thunk.ax>
   0x08048592 <+12>:    add    eax,0x1a6e
   0x08048597 <+17>:    sub    esp,0xc
   0x0804859a <+20>:    lea    edx,[eax-0x16f0]
   0x080485a0 <+26>:    push   edx
   0x080485a1 <+27>:    mov    ebx,eax
   0x080485a3 <+29>:    call   0x8048420 <system@plt>
   0x080485a8 <+34>:    add    esp,0x10
   0x080485ab <+37>:    nop
   0x080485ac <+38>:    mov    ebx,DWORD PTR [ebp-0x4]
   0x080485af <+41>:    leave
   0x080485b0 <+42>:    ret
End of assembler dump.
```

GDB

Reverse Engineering Weapons



```
loc_80484D0:
mov     eax, [eax+4]
add     eax, 4
mov     eax, [eax]
sub     esp, 8
push    offset aSuperSecretPas ; "super_secret_password"
push    eax
call    _strcmp
add     esp, 10h
test    eax, eax
jz      short loc_8048504
```

```
sub     esp, 0Ch
push    offset aAccessDenied ; "Access denied."
call    _puts
add     esp, 10h
```

```
loc_8048504:
sub     esp, 0Ch
push    offset aAccessGranted ; "Access granted."
```

```
80.00% (641,791) (1226,98) 0000049B 000000000804849B: main (Synchronized with Hex View-1)
```

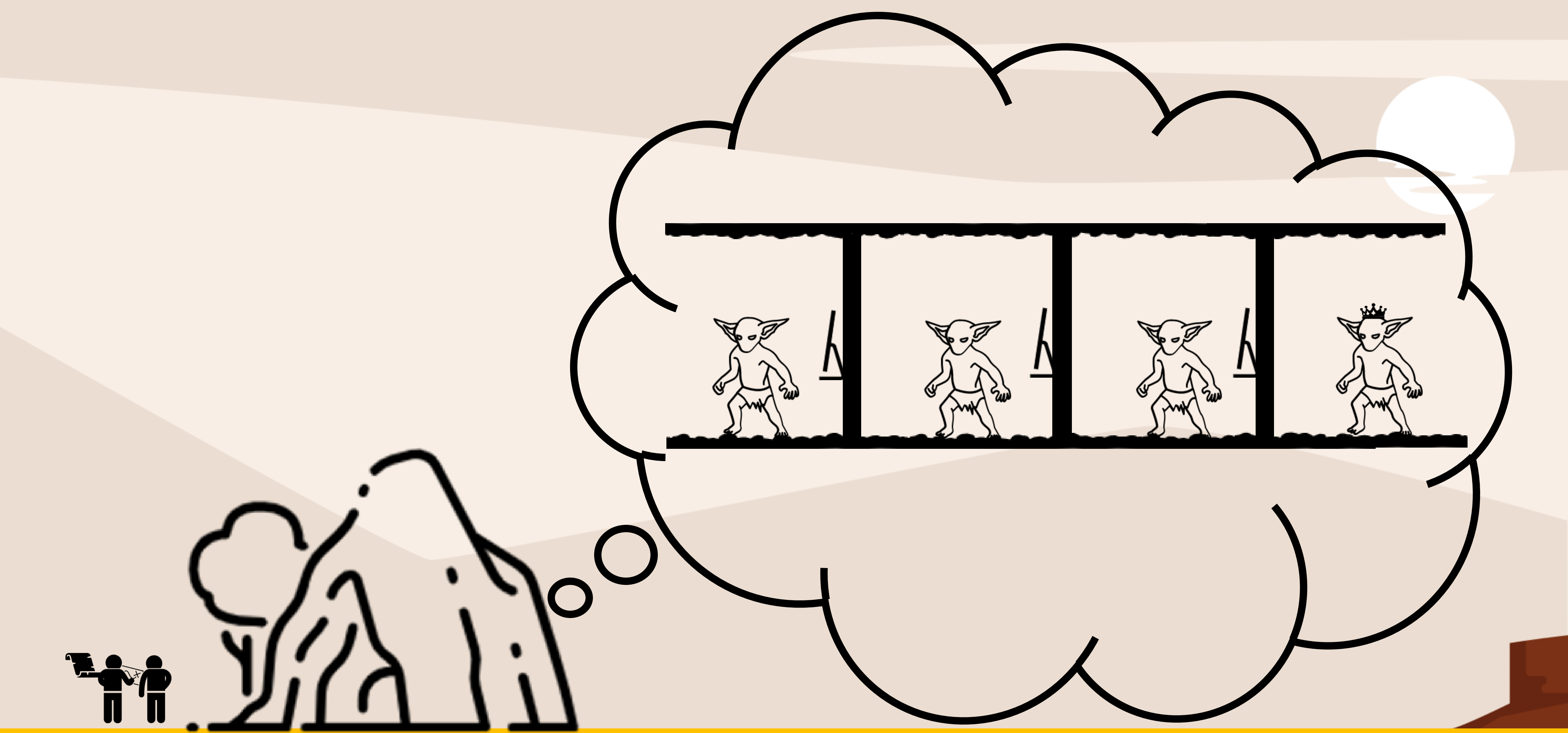
```
The initial autoanalysis has been finished.
```

```
AU:  idle   Down     Disk: 5GB
```

IDA

Solving the challenges at Sentinel Beacon, you move on to Enigma Caverns.

Upon reaching, you hear cackling sounds coming deep inside.



There was a goblins' den inside the cave. Luckily, the Whispering Parchment had updated itself.

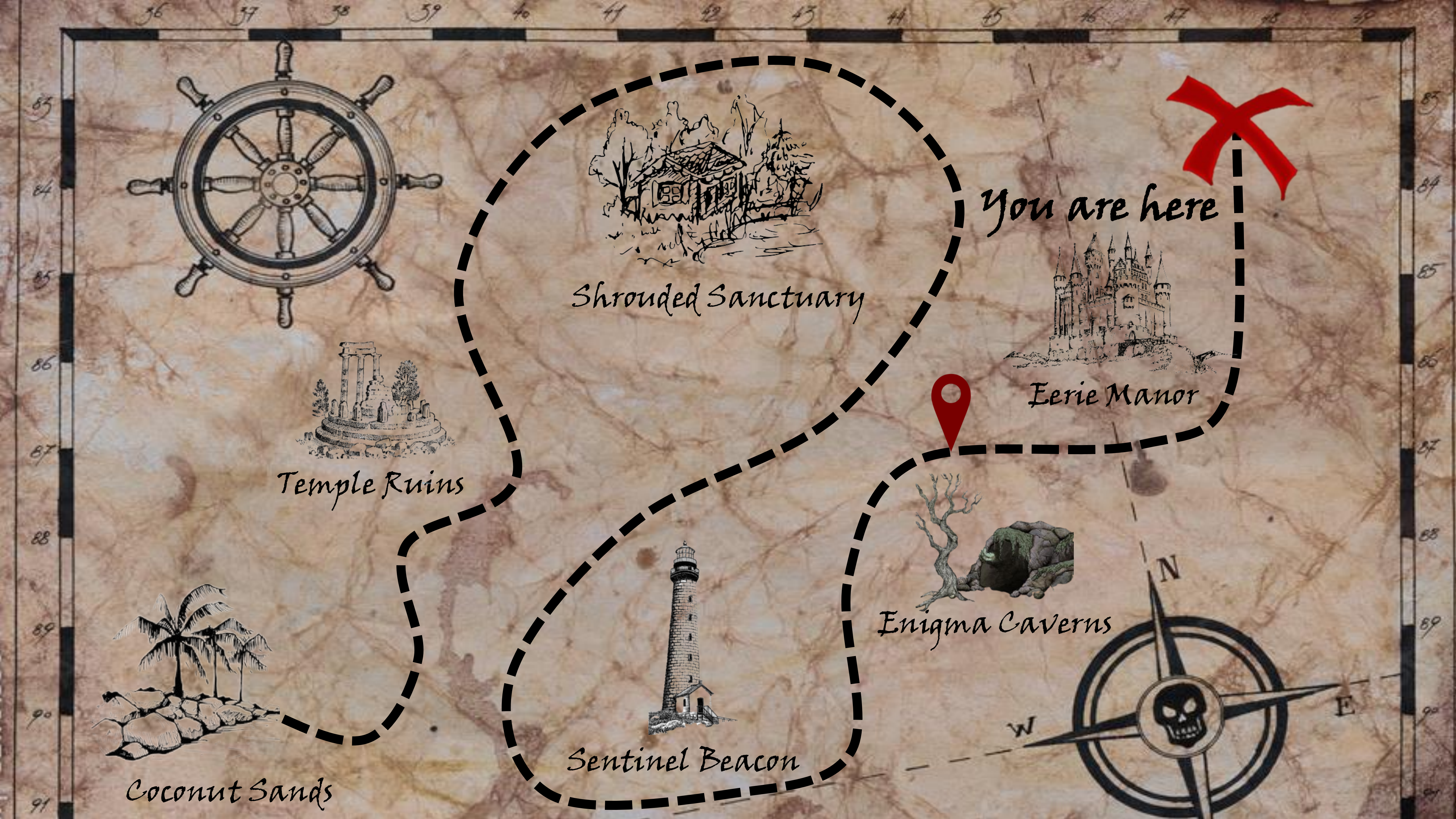
Three guarded doors, a goblin's lair,
Challenges to solve, if you dare.

Reverse engineering, the key to succeed,
To unravel the secrets, and the doors will concede,

The treasure awaits, but first you must see,
The goblin king, who holds the key.

Task:
Solve 3 reverse engineering challenges to unlock the sealed doors and meet the goblin king to reveal the next location.

HANDS-ON ACTIVITIES


You are here
Shrouded Sanctuary
Eerie Manor
Temple Ruins
Enigma Caverns
Sentinel Beacon
Coconut Sands
N
E
W

BINARY EXPLOITATION

Binary Exploitation (BE)

CAUTION: DIFFICULT CTF CATEGORY

What is Binary Exploitation in CTF?
Reverse engineer a program to find vulnerabilities in it and break the program.

Common Binary Exploitation challenges:

- Buffer Overflow
- Stack Overflow
- Heap Overflow
- Format String Vulnerabilities

Tools
Similar to Reverse Engineering

Difference between BE and RE

Binary Exploitation

Changing the behavior of a program to do something that it was not supposed to do.

Reverse Engineering

Understand how a program works.

BE requires some RE **but** RE does not necessarily involve BE.

At last, you have reached Eerie Manor where ghosts wander its corridors.

As usual, the Whispering Parchment has the new instructions you need...

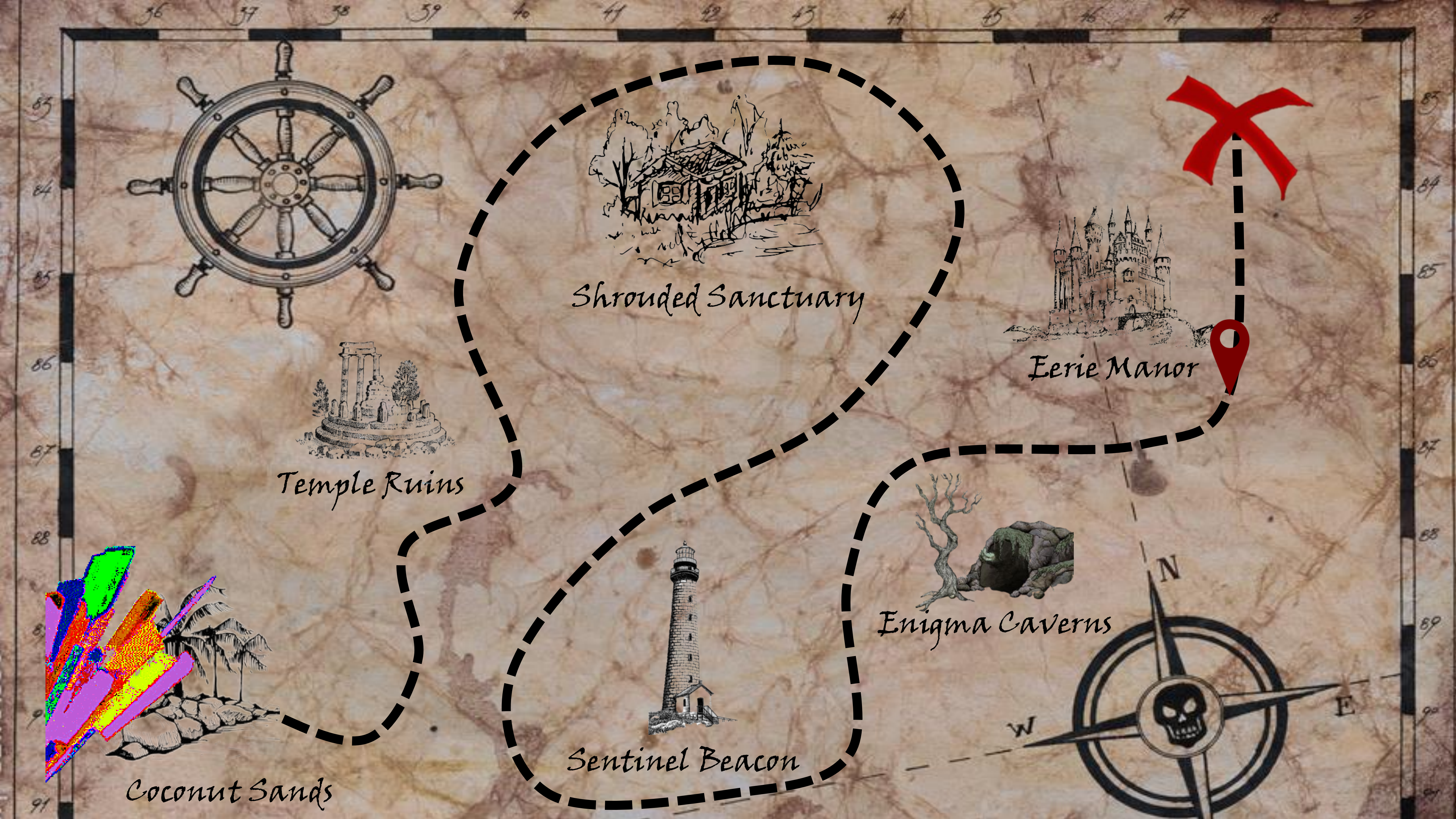
In Eerie Manor, where ghostly whispers speak,
A challenge awaits, for those who seek

A computer game, it asks of thee
Rock, paper, scissors, the challenge be

Outwit, outplay, and win the game
And the location of the treasure, shall be your claim

Task: Beat the ghost's computer in rock, paper, scissors

HANDS-ON ACTIVITIES


Shrouded Sanctuary
Eerie Manor
Temple Ruins
N
Enigma Caverns
E
W
Sentinel Beacon
Coconut Sands

BREAK TIME!