APU IBOH CTF WRITE-UP

Write-up by N3WBEES

Ariff | Iffat | Nawfal

TABLE OF CONTENT

[REV] ReverseMe

✨ Walkthrough by Ariff @ Rydzze

Link to Write-up: ReverseMe

[PWN] SHanity Checker

✨ Walkthrough by Ariff @ Rydzze

Link to Write-up: SHanity Checker

[PWN] Favourite Shell

✨ Walkthrough by Ariff @ Rydzze

Link to Write-up: Favourite Shell

[WEB] Echoes of the System

✨ Walkthrough by Iffat

When opening the website, you will be greeted at **index.html**. Go through all the pages and at **feedback.html**, there is a text form where you can insert your feedback. It seems to be able to inject payload.

Try injecting using `cat flag` but it shows error as it has sanitized the keywords **cat** and **flag**.

When injecting `ls -l`, it shows all the files name that included in the system.

There is the flag.txt file but we cannot use keywords flag to print out the file.

Final try injecting using `strings *.txt` and results in getting the flag.

🏳 Flag

`IBOH24{G3T_DaA8880948_F1aG}`

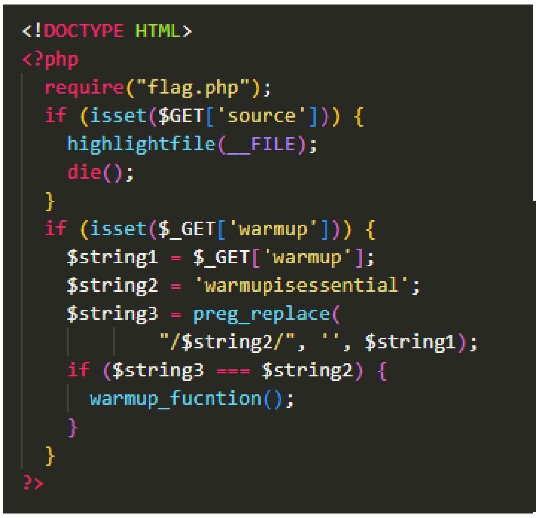
[WEB] Warmup

✨ Walkthrough by Iffat

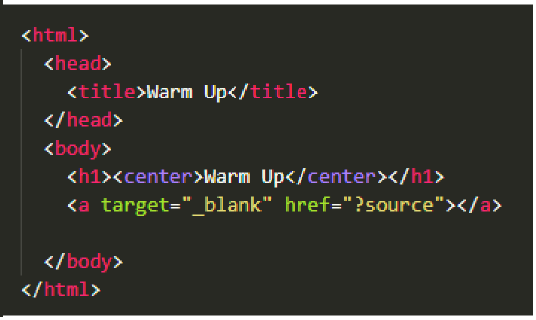
When opening the website, you will be greeted with blank page contains “Warm Up” text.

See the page source, and there is some clue when you include ?source inside the website link.

Clue:



```
<!DOCTYPE HTML>
<?php
  require("flag.php");
  if (isset($GET['source'])) {
    highlightfile(__FILE);
    die();
  }
  if (isset($_GET['warmup'])) {
    $string1 = $_GET['warmup'];
    $string2 = 'warmupisessential';
    $string3 = preg_replace(
            "/$string2/", '', $string1);
    if ($string3 === $string2) {
      warmup_fucntion();
    }
  }
?>
```



```
<html>
  <head>
    <title>Warm Up</title>
  </head>
  <body>
    <h1><center>Warm Up</center></h1>
    <a target="_blank" href="?source"></a>

  </body>
</html>
```

Basically, the code will get what we put in the link “?warmup=anything” (in this case it will get anything and insert into $string1) and it will compare with $string2.

> If there is $string2 inside the $string1, it will replace with ' ' which it will remove the same string as in $string2 and insert into $string3.
>
> If $string3 is equal in value and type (===) with $string2, it will call the warmup_fucntion(); (*yes, it was misspelled*) and maybe it will print the flag that we have been searching for.

So, for the solution, I try to let the link as below:

`?warmup=warmupiswarmupisessentialessential`

It will show us the flag.

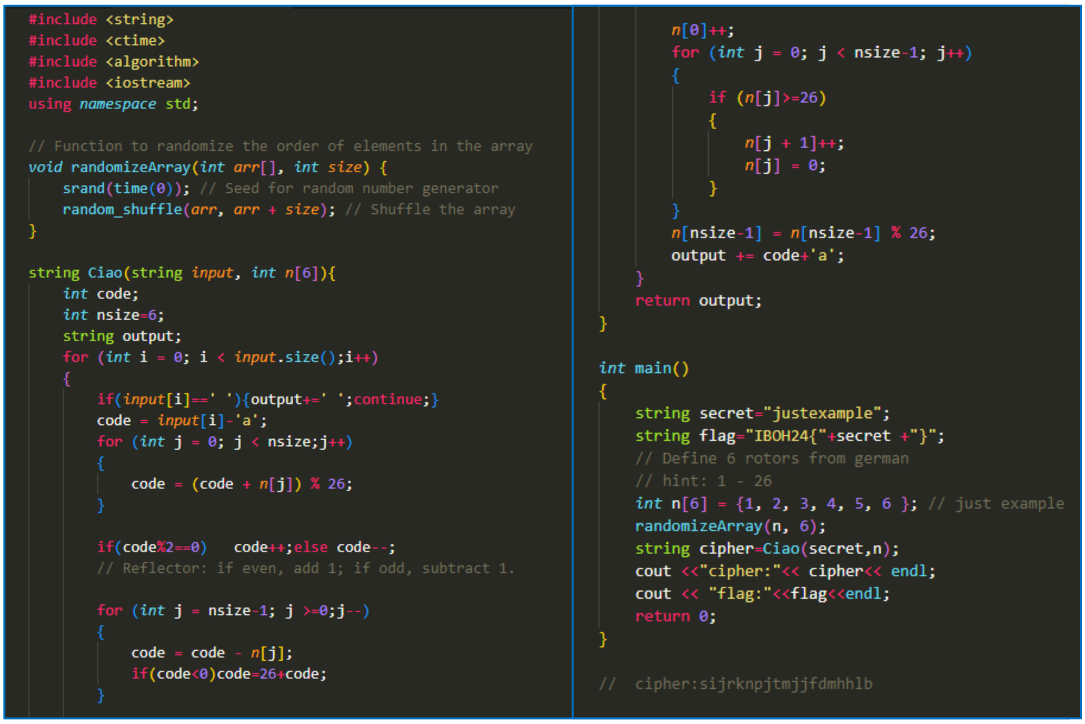
🏳 Flag

`IBOH24{5e83215e5db52738f7699a3c5d94702c}`

[CRYPTO] Grub Gott

✨ Walkthrough by Nawfal

Given a C++ source code as shown below:



```cpp
#include <string>
#include <ctime>
#include <algorithm>
#include <iostream>
using namespace std;

// Function to randomize the order of elements in the array
void randomizeArray(int arr[], int size) {
    srand(time(0)); // Seed for random number generator
    random_shuffle(arr, arr + size); // Shuffle the array
}

string Ciao(string input, int n[6]){
    int code;
    int nsize=6;
    string output;
    for (int i = 0; i < input.size();i++)
    {
        if(input[i]==' '){output+=' ';continue;}
        code = input[i]-'a';
        for (int j = 0; j < nsize;j++)
        {
            code = (code + n[j]) % 26;
        }

        if(code%2==0)   code++;else code--;
        // Reflector: if even, add 1; if odd, subtract 1.

        for (int j = nsize-1; j >=0;j--)
        {
            code = code - n[j];
            if(code<0)code=26+code;
        }
```

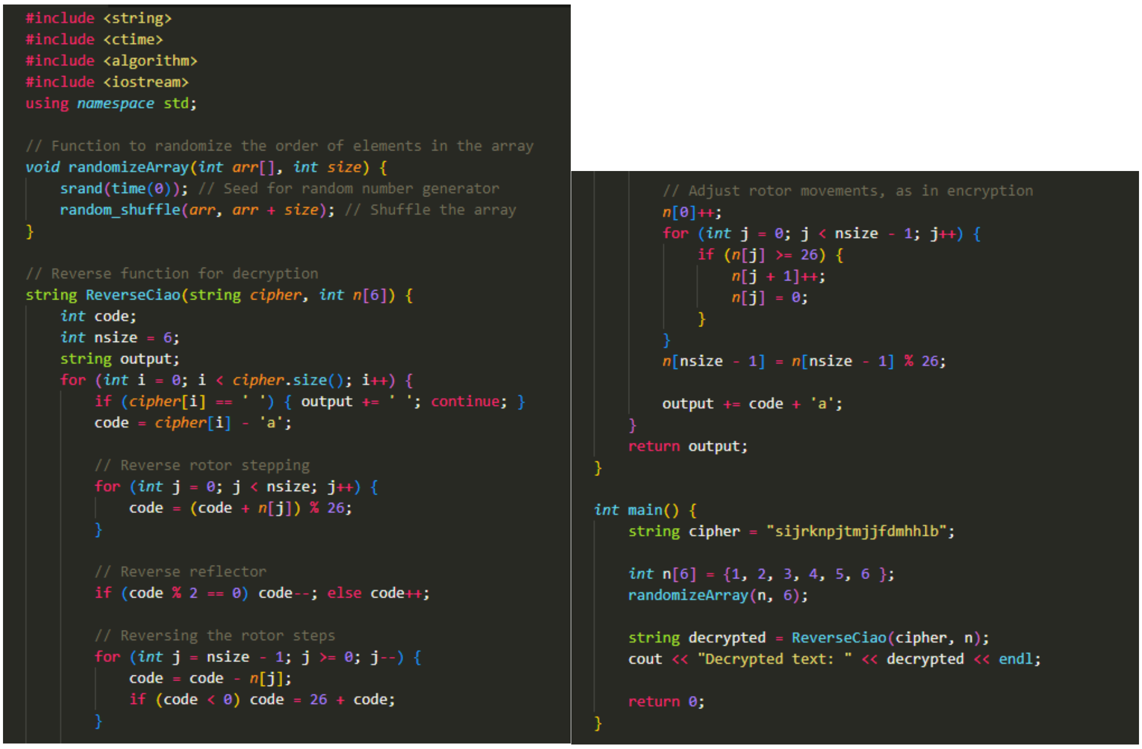
```cpp
            n[0]++;
            for (int j = 0; j < nsize-1; j++)
            {
                if (n[j]>=26)
                {
                    n[j + 1]++;
                    n[j] = 0;
                }
            }
            n[nsize-1] = n[nsize-1] % 26;
            output += code+'a';
        }
        return output;
    }

    int main()
    {
        string secret="justexample";
        string flag="IBOH24{"+secret +"}";
        // Define 6 rotors from german
        // hint: 1 - 26
        int n[6] = {1, 2, 3, 4, 5, 6 }; // just example
        randomizeArray(n, 6);
        string cipher=Ciao(secret,n);
        cout <<"cipher:"<< cipher<< endl;
        cout << "flag:"<<flag<<endl;
        return 0;
    }

    //  cipher:sijrknpjtmjjfdmhhlb
```

This code implements a rotor-based substitution cipher similar to the Enigma machine. It uses six rotors (numbers in an array) to shift each letter of the input string forward and backward, wrapping around the alphabet. A reflector step adjusts each letter: if the shifted value is even, it's incremented by 1; if odd, it's decremented by 1. After processing each letter, the rotors increment to change the shift for the next character. The rotors are randomized at the start, making the cipher different each time the program is run.

For the decryption process, just change at the if-else statement of reflector.

If even, subtract 1; if odd, add 1

Script to Decrypt the Cipher Text



```cpp
#include <string>
#include <ctime>
#include <algorithm>
#include <iostream>
using namespace std;

// Function to randomize the order of elements in the array
void randomizeArray(int arr[], int size) {
    srand(time(0)); // Seed for random number generator
    random_shuffle(arr, arr + size); // Shuffle the array
}

// Reverse function for decryption
string ReverseCiao(string cipher, int n[6]) {
    int code;
    int nsize = 6;
    string output;
    for (int i = 0; i < cipher.size(); i++) {
        if (cipher[i] == ' ') { output += ' '; continue; }
        code = cipher[i] - 'a';

        // Reverse rotor stepping
        for (int j = 0; j < nsize; j++) {
            code = (code + n[j]) % 26;
        }

        // Reverse reflector
        if (code % 2 == 0) code--; else code++;

        // Reversing the rotor steps
        for (int j = nsize - 1; j >= 0; j--) {
            code = code - n[j];
            if (code < 0) code = 26 + code;
        }
        // Adjust rotor movements, as in encryption
        n[0]++;
        for (int j = 0; j < nsize - 1; j++) {
            if (n[j] >= 26) {
                n[j + 1]++;
                n[j] = 0;
            }
        }
        n[nsize - 1] = n[nsize - 1] % 26;

        output += code + 'a';
    }
    return output;
}

int main() {
    string cipher = "sijrknpjtmjjfdmhhlb";

    int n[6] = {1, 2, 3, 4, 5, 6 };
    randomizeArray(n, 6);

    string decrypted = ReverseCiao(cipher, n);
    cout << "Decrypted text: " << decrypted << endl;

    return 0;
}
```

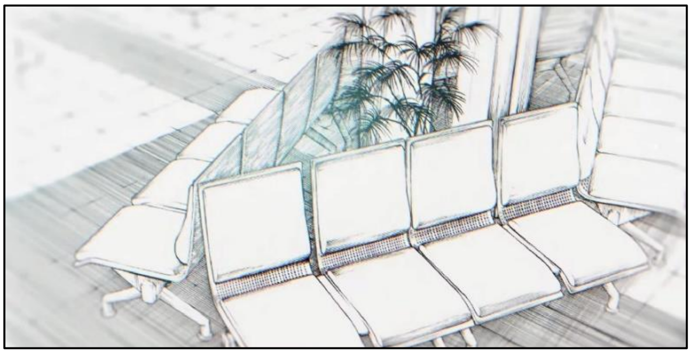
Flag

`IBOH24{thislookslikeenigma}`

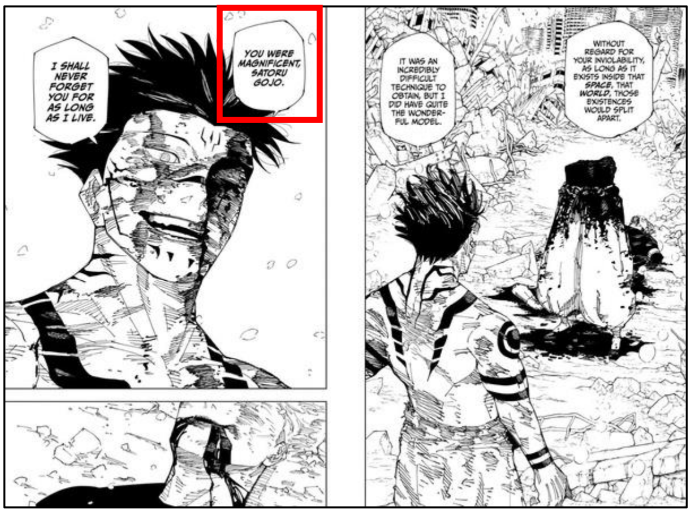
[OSINT] You Were Magnificent

✨ Walkthrough by Nawfal

The image given:



Basically, the image can be import into Google Images to find the source which use related image with the given one. After that, we need to do some information digging which conclude that the image is taken from a Manga called Jujutsu Kaisen in Chapter 236 which represents the moment Gojo Satoru got *offscreened*. After a thoroughly search I found out that a [Reddit discussion](#) mentioned the name of the airport which is **Haneda Airport** located in Tokyo.


WITHOUT REGARD FOR YOUR INVIOLABILITY, AS LONG AS IT EXISTS INSIDE THAT SPACE, THAT WORLD, THOSE EXISTENCES WOULD SPLIT APART.
IT WAS AN INCREDIBLY DIFFICULT TECHNIQUE TO OBTAIN, BUT I DID HAVE QUITE THE WONDERFUL MODEL.
YOU WERE MAGNIFICENT, SATORU GOJO.
I SHALL NEVER FORGET YOU FOR AS LONG AS I LIVE.

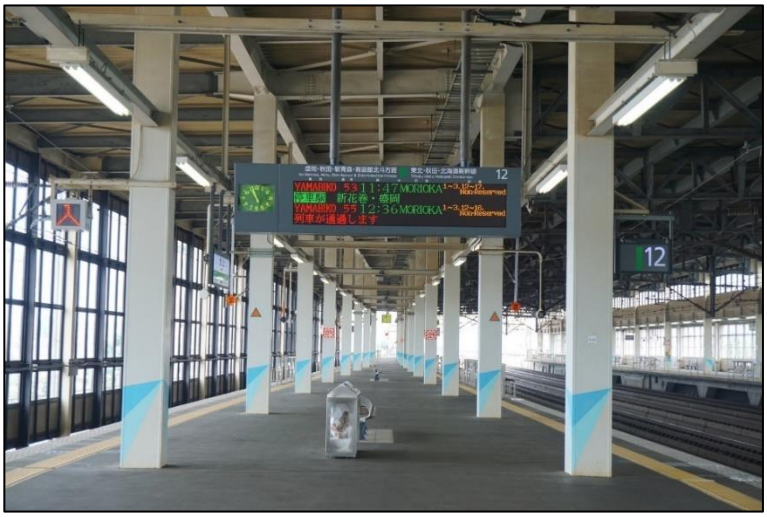
🏳 Flag

`IBOH24{Haneda_Airport}` *(not sure, we forgor shshsh)*

[OSINT] Ryōiki Tenkai

✨ Walkthrough by Nawfal

The image given:


盛岡・秋田・新青森・新函館北斗方面
東北・秋田・北海道新幹線
12
YAMABIKO 53 11:47 MORIOKA 1~3,12~17, Non-Reserved
停車駅 新花巻・盛岡
YAMABIKO 55 12:36 MORIOKA 1~3,12~16, Non-Reserved
列車が通過します
12

Basically, the image can be import into Google Images to find the source which use related image with the given one. After that, we need to do some information digging which conclude that the image is taken from a train station named Kitakami Station located in the city of Kitakami, Iwate, Japan. Therefore, I search the [Wikipedia of the train]() and noticed that the wiki includes the train line(s) associated with the train station. But the problem is when the I enter the lines according to the sequences given by the wiki in the flag turns out to be wrong and therefore, I tried every combination to relocate the lines sequence and turns out to be reversed.

WHAT THE HELL IS THIS PLACE...?
10

🏳 Flag

`IBOH24{KitakamiLine_ TōhokuMainLine_ TōhokuShinkansenLine}`

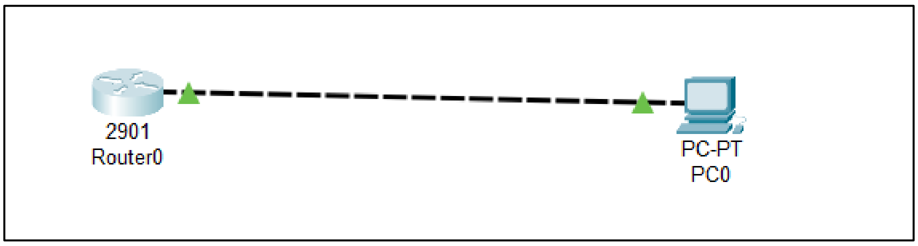
[MISC] Welcome Back to Cisco

✨ Walkthrough by Iffat

In the challenge description, we are given the SSH credentials for us to login to the router inside the Packet Tracer file that are also given.

SSH Credentials:

IP Address: 192.168.69.1
Username: guest
Password: IBOH24



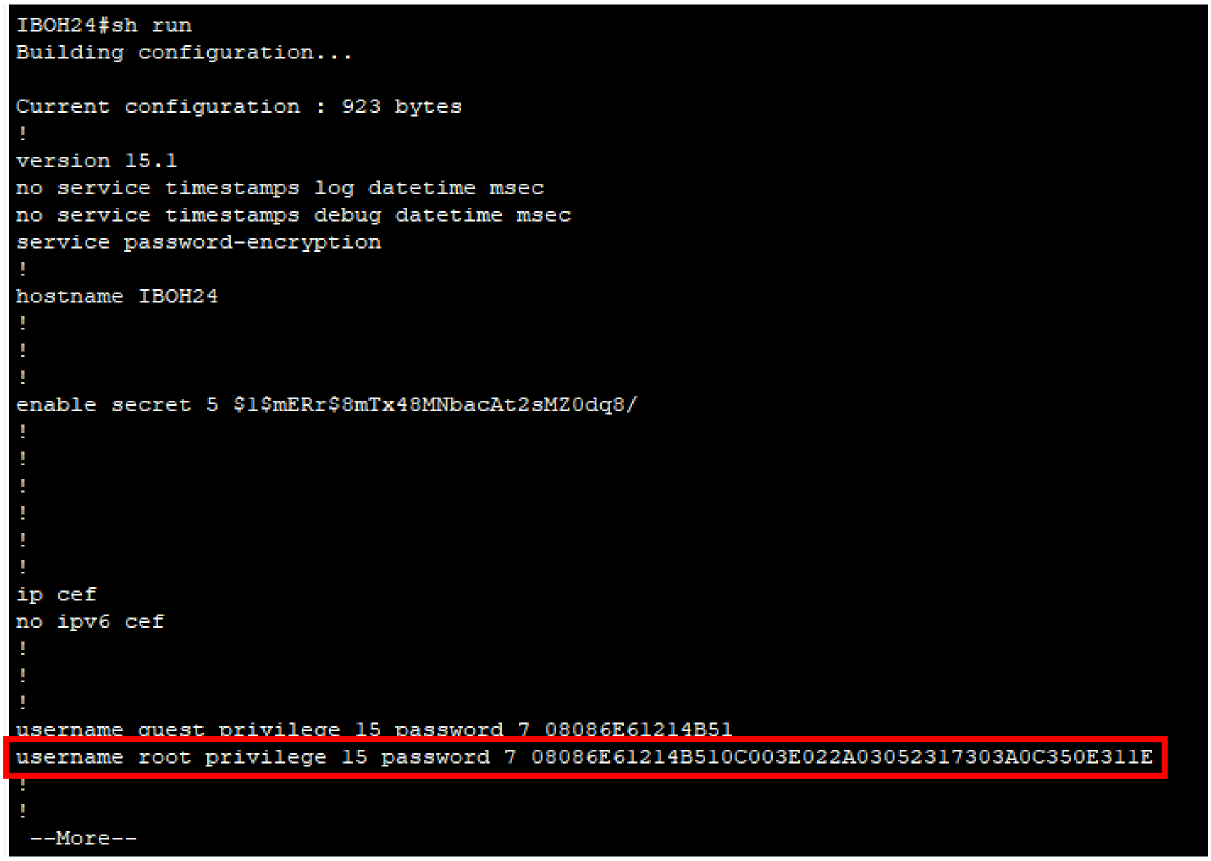
Go to the PC0 and at Desktop Tab, click the Telnet/SSH Client. Login using the credentials given.

Try to check the router configuration using the command:

```
IBOH24# sh run
```

Usually, all the configurations including the password will be print out using the command above.

We need to get the root password from the router, click enter until you found the root password.



```
IBOH24#sh run
Building configuration...

Current configuration : 923 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname IBOH24
!
!
!
enable secret 5 $1$mERr$8mTx48MNbacAt2sMZ0dq8/
!
!
!
!
!
!
ip cef
no ipv6 cef
!
!
!
username guest privilege 15 password 7 08086E61214B51
username root privilege 15 password 7 08086E61214B510C003E022A03052317303A0C350E311E
!
!
 --More--
```

The password seems to be encrypted, open web browser and find Cisco Password 7 decrypt

I used this website to decrypt the password: https://www.ifm.net.nz/cookbooks/passwordcracker.html

Just copy and paste the root password and you will get the flag.

Flag

`IBOH24{rUnNINg_coNFiG}`