A screenshot of a game

Description automatically generated

This challenge is situated under the Web category. A beta version of a Slot Machine game has been created at this website and it contains a bug. The flag is rewarded once the bug has been found and exploited. Hints given are as follows:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a slot machine

Description automatically generated

Upon clicking the website link, the player the greeted with the slot machine. The slot machine consists of the slots, Credits amount, Bet amount and ‘SPIN’ button. Based on the description, the player needs to get 1,000,000,000 to win and get the flag.

Spinning the slot machine and it can be observed as follows:

A screenshot of a slot machine

Description automatically generated

A losing spin with a bet of 1. The credits are deducted by 1 from 1000 to 999.

A screenshot of a slot machine

Description automatically generated

A winning spin with a bet of 25. The winning is double and raising the credit to 984.

Reaching to 1 billion will take a long time if the usual route of just spinning the slot machine is taken. Hence, as stated in the challenge description the bug will the searched for and exploited in order to obtain the flag.

1. Viewing the source file

A screen shot of a computer program

Description automatically generated

First, the page source of the slot machine is viewed. By examining the source file, it can be seen that the website is a simple slot machine game with a credit system. The game allows users to spin the reels and win credits based on the outcome. There's a condition in the JavaScript code that if the player's credit reaches 1,000,000,000, a flag will be revealed. This HTML file contains the structure and layout for the slot machine game, while the external JavaScript and CSS files are used to add interactivity and styling to the game.

1. Examining the JavaScript code

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

The JavaScript code (slotEngine.js) contains the logic for the slot machine game. By examining the code, it is observed that the game spins three reels with random numbers. It is also apparent that if all three reels have the same number, the credit is increased by three times the bet amount. If two reels have the same number, the credit is increased by two times the bet amount. Otherwise, the bet amount is deducted from the credit. When the player wins and their credit reaches 1,000,000,000, an AJAX request is sent to a server-side PHP script (secret.php) with the player's credit as a parameter and presumably reveals the flag. However, the credit or bet amount cannot be modified directly through the JavaScript code.

1. Bug Exploitation

The vulnerability lies in the insufficient validation of the price variable before triggering the AJAX request. The game only checks if the player's credits (price) are greater than or equal to 1,000,000,000 to trigger the flag. There's no validation to ensure that the credits are obtained legitimately or that the value is within reasonable bounds.

A screen shot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

The condition if(price >= 1000000000) in this area of the code, which determines whether the player's credits (price) have surpassed 1,000,000,000, is where the vulnerability is. AJAX request to reveal the flag is made to a server-side script (secret.php) if this condition evaluates to true.

The player's credits aren't properly enforced or validated, which is the bug. Players could be able to set the price variable to 1,000,000,000 or more without really earning or rightfully gaining that number of credits through gameplay because it is being modified directly on the client-side. Without fulfilling the intended gaming requirements, players can take advantage of the game's lack of validation and trigger the flag-revealing condition.

Exploiting this vulnerability can be done in multiple ways:

* Manipulating the client-side credits directly to reach 1,000,000,000.
* Manipulating the bet amount to cause the player's credits to go below 0 or above 1,000,000,000.

Method 1: Python Code

A screenshot of a computer program

Description automatically generated

1. import requests

This line imports the requests library, which is used to send HTTP requests in Python.

1. url = "https://skrctf.me/ports/32d3f74be4bf0b6c2de4862cbcf9aa8a/secret.php"

The flag can be acquired from this line, which specifies the URL of the vulnerable website.

1. payload = {"price": "1000000000"}

The payload is specified here, and it will be delivered with the POST request. In the payload, there is only one key-value pair: "price" is the key, and "1000000000" is the value, which stands for the fabricated credit value.

1. response = requests.post(url, data=payload)

Using the payload (payload) that is supplied, this line makes a POST request to the given URL (url). You can send a POST request using the requests.post() function, and you can define the data to be delivered in the request body using the data parameter.

1. if response.status\_code == 200:

print("Flag:", response.text)

else:

print("Failed to retrieve the flag.")

This is where the response's status code is verified with, 200, signifies a successful request. The output the flag the server returned (response.text) if the request was successful. If there was a problem with the server or connectivity, for example, and the request failed, the failure message will be printed out.

1. The script is run, and the flag is obtained.

Method 2: Console (Credit Manipulation)

In the code snippet below, variable ‘price’ represents the player’s credits. By directly manipulating the price variable on the client-side, the desired value of 1 billion can be set. This is done through the ‘Inspect Element’ and under the Console tab.

A screen shot of a computer program

Description automatically generated

The typical gameplay technique of collecting credits through spins is bypassed and the threshold needed to trigger the flag-revealing condition is reached by setting price to 1,000,000,000.

There's a condition that checks if the player's credits (price) are greater than or equal to 1,000,000,000. When this condition evaluates to true (which it will if price is set to 1,000,000,000), it executes the code inside the if statement block.

Inside the if statement block, an AJAX request is sent to a server-side script (secret.php) to reveal the flag. This code creates an XMLHttpRequest object (xhttp) and sends a POST request to the server-side script (secret.php) with the price parameter set to 1,000,000,000. If the request is successful (status code 200), the flag returned by the server is displayed in an alert box.

A screenshot of a computer

Description automatically generated

Method 3: Console (Bet Manipulation)

document.getElementById("bets").innerHTML = "-100000000000000";

The line above is a JavaScript statement that manipulates the content of an HTML element with the id attribute set to "bets"

A screenshot of a computer

Description automatically generated

1. document.getElementById("bets"):

* ‘document’ refers to the Document Object Model (DOM) of the HTML document currently loaded in the browser.
* ‘getElementById("bets")’ is a DOM method used to retrieve an HTML element with a specific id attribute value.
* In this case, it's looking for an element with the id "bets".

1. .innerHTML:

* innerHTML is a property of DOM elements that represents the HTML content within the element, including any nested elements.
* By setting this property, you can change the HTML content of the selected element.

1. = "-100000000000000";:

* This part of the statement assigns a new value to the innerHTML property of the element with id "bets".
* The assigned value is a string "-100000000000000".

A screenshot of a computer

Description automatically generated

The line paper, which summarises everything.getElementById(); say "bets"innerHTML = "-100000000000000"; replaces the content of an HTML element with the id "bets" with the string "-100000000000000". By doing this, the element's visible text or HTML content is essentially changed to "-100000000000000".

A screenshot of a game

Description automatically generated