

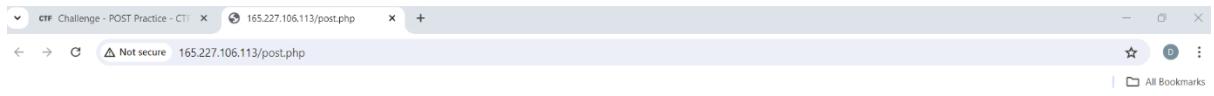
CFSS PAID Internship Program

Task 1 : <https://ctflearn.com/challenge/114>

Step 1 : First Go to this website.

The screenshot shows a web browser window with the URL ctflearn.com/challenge/114. The page title is "Challenge - POST Practice - CTF". The main content area displays a challenge titled "POST Practice" with 40 points and a medium difficulty level. The description states: "This website requires authentication, via POST. However, it seems as if someone has defaced our site. Maybe there is still some way to authenticate? <http://165.227.106.113/post.php>". Below this, there is a form with a "Flag" input field containing "CTFlearn[h4ck3d]" and a "Submit" button. To the right of the challenge details, there is a "Top10" scoreboard table and a "Rating" chart. The "Top10" table lists 10 users with their ratings: 1. ross3102, 2. alexkato29, 3. emperorlepone, 4. dknj11902, 5. Oxibrain, 6. thanhbok26b, 7. voidmercy, 8. nicev20, 9. limyunkai19, and 10. nandayo. The "Rating" chart shows a distribution of 5-star, 4-star, 3-star, 2-star, and 1-star ratings. The bottom of the browser window shows the Windows taskbar with various pinned icons and system status indicators.

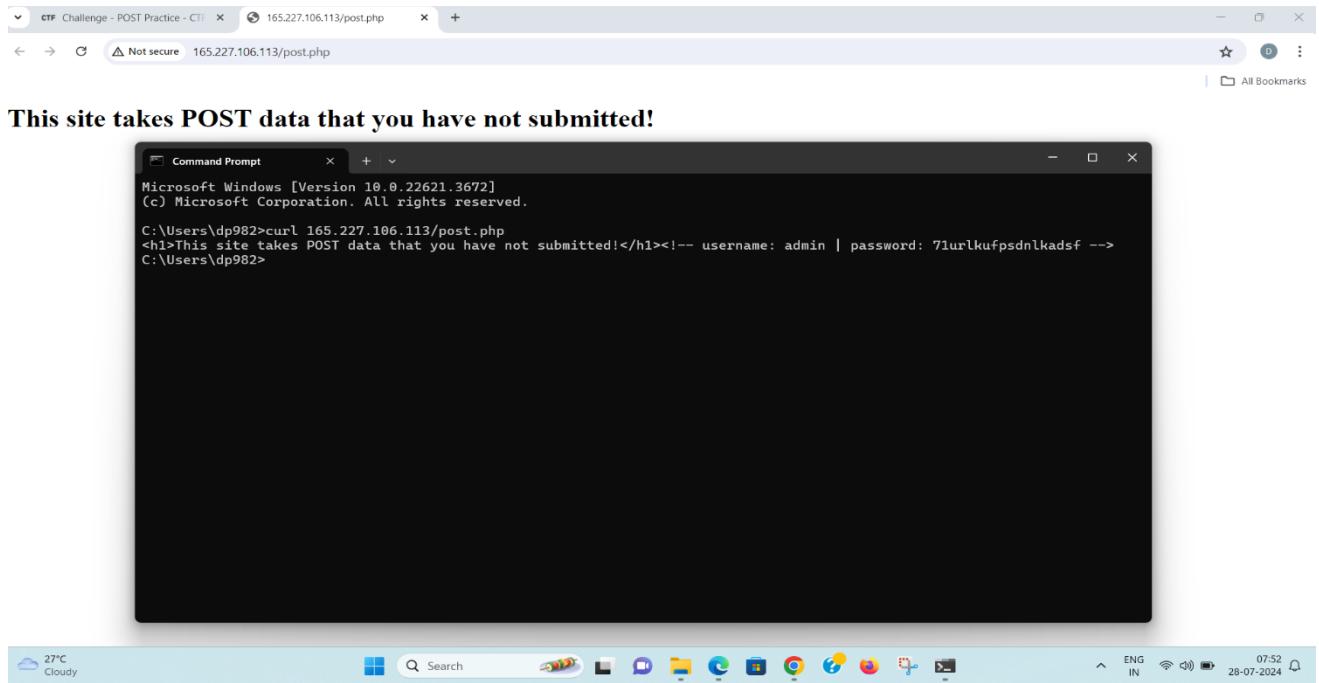
Step 2 : Go to <http://165.227.106.113/post.php>



You can see the page like this.

Step 3 : Open cmd and type command -

curl 165.227.106.113/post.php & press enter



You can show the username and password ,this is the first trick of that using curl you can get the username and password.

or

Step 3 : Using Inspect elements you can know username and password.

You can see that in body tag.

A screenshot of a web browser window titled "Challenge - POST Practice - CTF". The address bar shows the URL "165.227.106.113/post.php". The page content is a single line of text: "This site takes POST data that you have not submitted!". Below the browser window is the developer tools interface. The "Elements" tab is selected, showing the HTML structure of the page. The "Styles" tab shows the CSS rules for the body element, including "display: block;" and "margin: 8px;". A green box highlights the margin rule. The browser's status bar at the bottom shows the date and time as "28-07-2024 07:53".

Step 4 : In cmd type command –

curl 165.227.106.113/post.php -d username=admin -d password=71urlkufpsdnlkadsf and press enter

A screenshot of a web browser window titled "Challenge - POST Practice - CTF". The address bar shows the URL "165.227.106.113/post.php". A command prompt window is overlaid on the browser, showing the output of a curl command. The command is: "curl 165.227.106.113/post.php -d username=admin -d password=71urlkufpsdnlkadsf". The output shows the page content "This site takes POST data that you have not submitted!" followed by the flag "flag{p0st_d4t4_4ll_d4y}".

The browser's developer tools interface is visible below the browser window. The "Elements" tab is selected, showing the HTML structure of the page. The "Styles" tab shows the CSS rules for the body element, including "display: block;" and "margin: 8px;". A green box highlights the margin rule. The browser's status bar at the bottom shows the date and time as "28-07-2024 07:58".

You can see the Flag.

Step 5 : Enter the flag in text field and press submit button.

The screenshot shows a browser window for a challenge titled "POST Practice" on the CTFLearn platform. The challenge has 40 points and is categorized as Medium. The description states: "This website requires authentication, via POST. However, it seems as if someone has defaced our site. Maybe there is still some way to authenticate? http://165.227.106.113/post.php". Below the description is a text input field containing "Flag CTFLearn{p0st_d4t4_4ll_d4y}" and a "Submit" button. To the right of the challenge details is a "Top10" scoreboard table:

Rank	User	Points
1	ross3102	40
2	alexkato29	40
3	emperorlepone	40
4	dknj11902	40
5	0xibram	40
6	thanhbok26b	40
7	voidmercy	40
8	niclev20	40
9	limyunkai19	40
10	nandayo	40

Below the Top10 section is a "Rating" chart showing the distribution of ratings from 1 star to 5 stars. The chart indicates that most users have rated the challenge as 4 or 5 stars. The total rating is 4.44. At the bottom of the challenge page, it says "20344 solves" and "Must solve to rate". The browser's taskbar at the bottom shows various open tabs and system icons.

Step 6 : You can see we're solved the Lab.

ctf Challenge - POST Practice - CTF x 165.227.106.113/post.php x | +

ctflearn.com/challenge/114

Please verify your email. Click to resend.

CTFLEARN Learn Challenges Scoreboard Dashboard Learn++

POST Practice ✓ 40 points Medium

This website requires authentication, via POST. However, it seems as if someone has defaced our site. Maybe there is still some way to authenticate?
http://165.227.106.113/post.php

Flag CTFLearn[h4ck3d] Solved

Web · intelagent 20345 solves

Top10

Rank	User	Points
1	ross3102	40
2	alexkato29	40
3	emperorlepone	40
4	dknj11902	40
5	Oxibrain	40
6	thanhbok26b	40
7	voidmercy	40
8	niclev20	40
9	limyunkai19	40
10	nandayo	40

Rating - Please Rate 4.44

5★ 4★ 3★ 2★ 1★

★★★★★

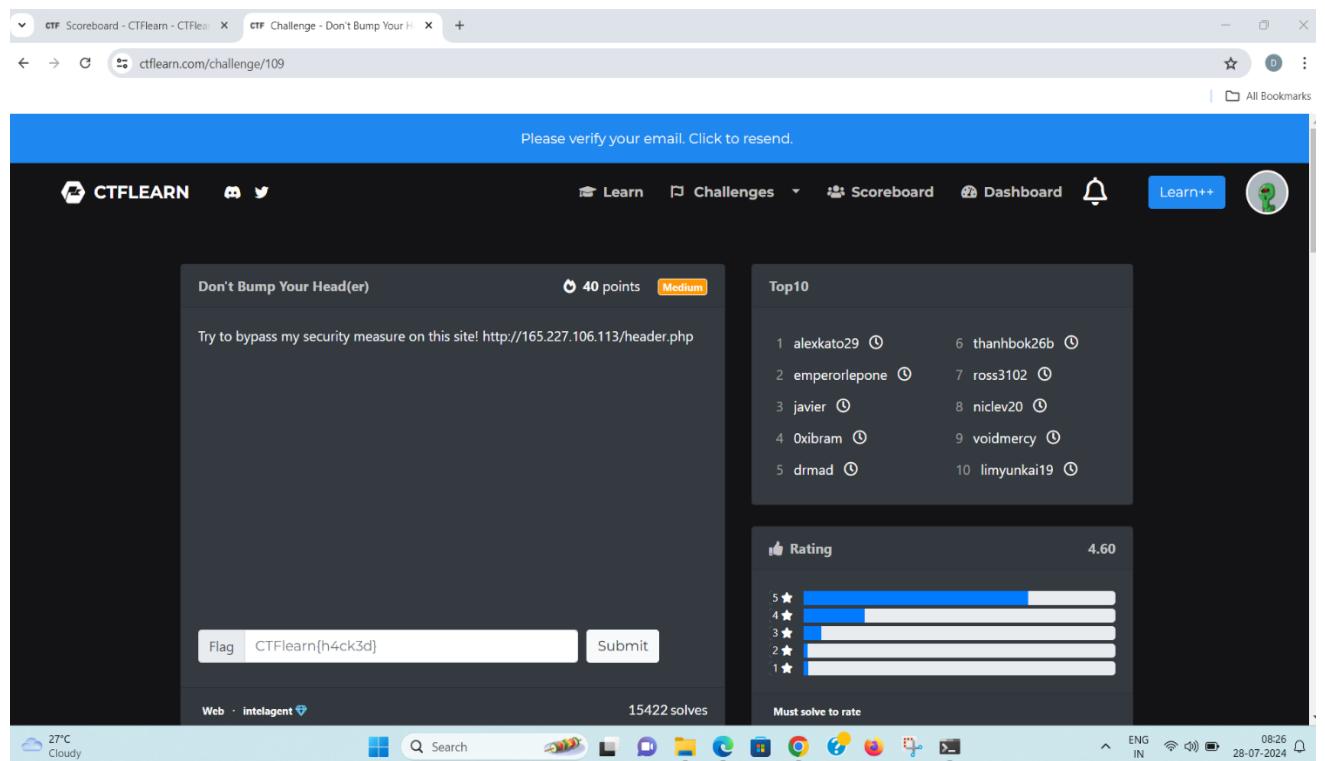
SL - IND Game score

Search

ENG IN 08:00 28-07-2024

Task 2 : <https://ctflearn.com/challenge/109>

Step 1 : First go to this website.



Step 2 : Go on this link :-

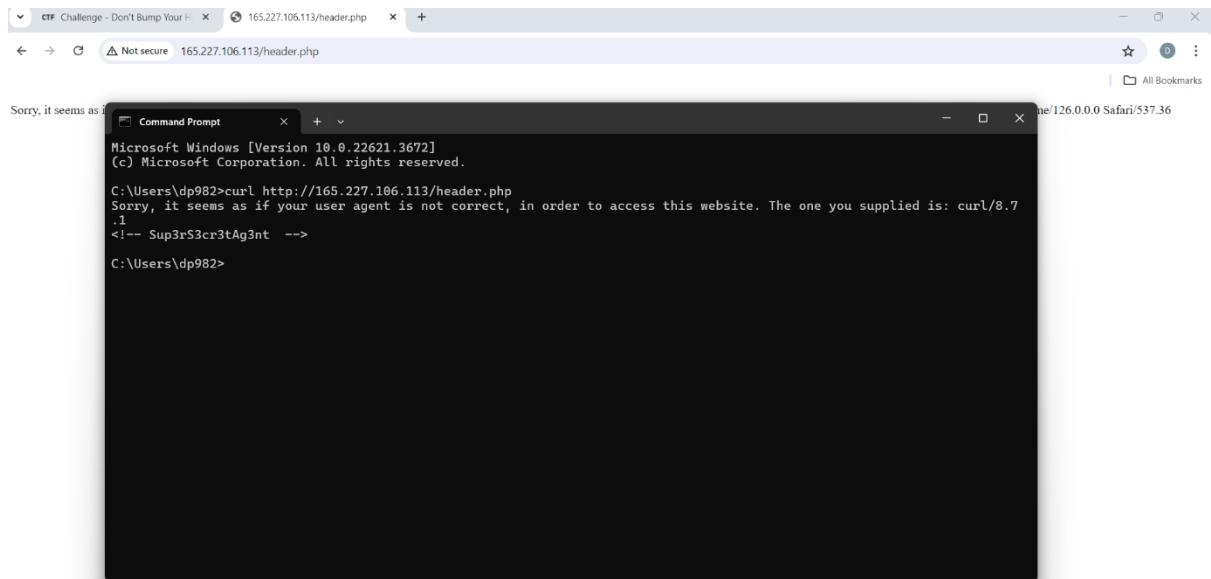
<http://165.227.106.113/header.php>



You can see the page like this.

Step 3 : Open cmd & type this command :-

```
curl http://165.227.106.113/header.php
```

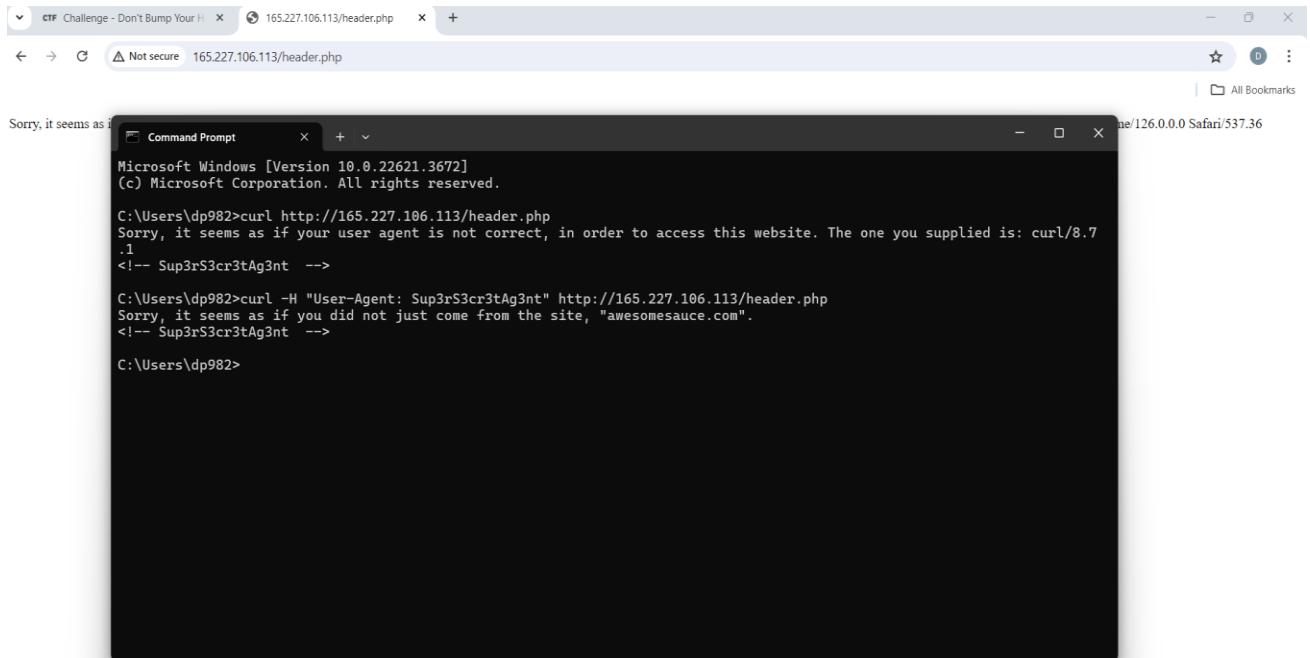


We got this “<! – SuperrS3cr3tAg3nt->”.

Step 4 : Next command :-

curl -H "User-Agent: Sup3rS3cr3tAg3nt"
<http://165.227.106.113/header.php>

for get the next hint or to head towards next step.

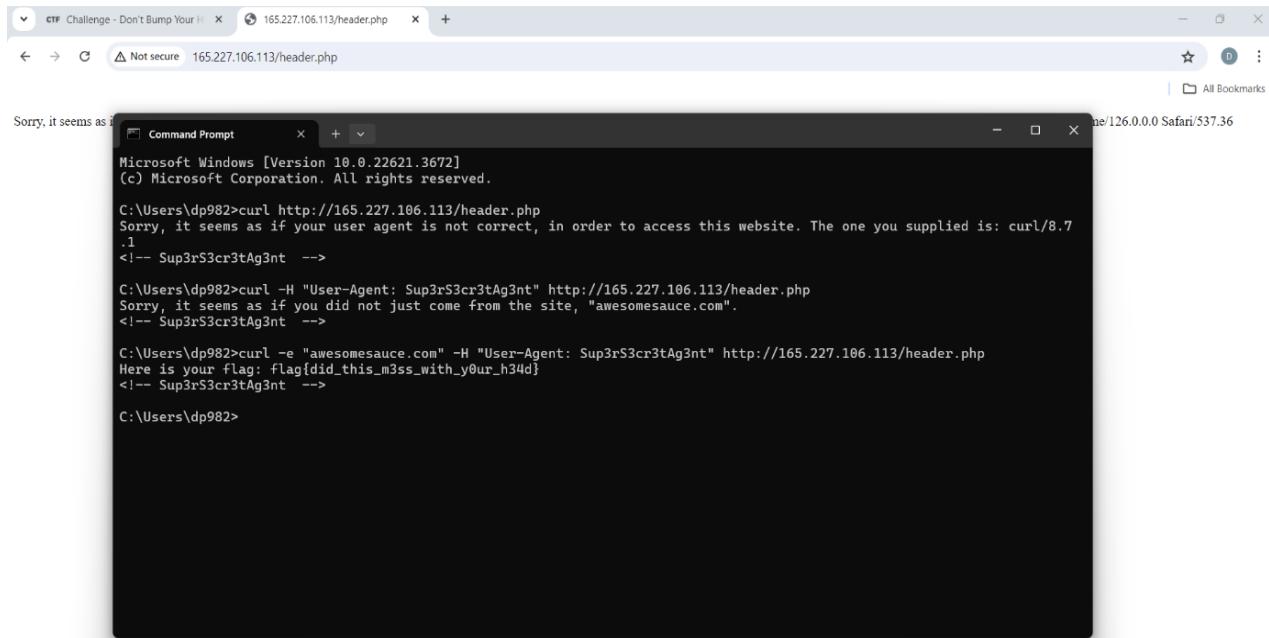


You can see output says “awesomesauce.com”.

Step 5 : Use this command :

```
>curl -e "awesomesauce.com" -H "User-Agent:
Sup3rS3cr3tAg3nt"
http://165.227.106.113/header.php
```

For referrer flag.



The screenshot shows a Windows desktop environment. At the top, there's a taskbar with various icons and a system tray showing the date and time (28-07-2024). In the center, a browser window is open to a page at 165.227.106.113/header.php, displaying a 'Not secure' warning. Below it, a Command Prompt window is running on Microsoft Windows Version 10.0.22621.3672. The user is executing curl commands to interact with the website. The output shows the user agent being set to 'Sup3rS3cr3tAg3nt' and receiving a response that includes the flag: 'flag{id_this_m3ss_with_y0ur_h34d}'.

```
Sorry, it seems as if your user agent is not correct, in order to access this website. The one you supplied is: curl/8.7
.C
<!-- Sup3rS3cr3tAg3nt -->
C:\Users\dp982>curl -H "User-Agent: Sup3rS3cr3tAg3nt" http://165.227.106.113/header.php
Sorry, it seems as if you did not just come from the site, "awesomesauce.com".
<!-- Sup3rS3cr3tAg3nt -->
C:\Users\dp982>curl -e "awesomesauce.com" -H "User-Agent: Sup3rS3cr3tAg3nt" http://165.227.106.113/header.php
Here is your flag: flag{id_this_m3ss_with_y0ur_h34d}
<!-- Sup3rS3cr3tAg3nt -->
C:\Users\dp982>
```

You can see it's captured the flag.

Step 6 : Copy the flag and press the submit button.

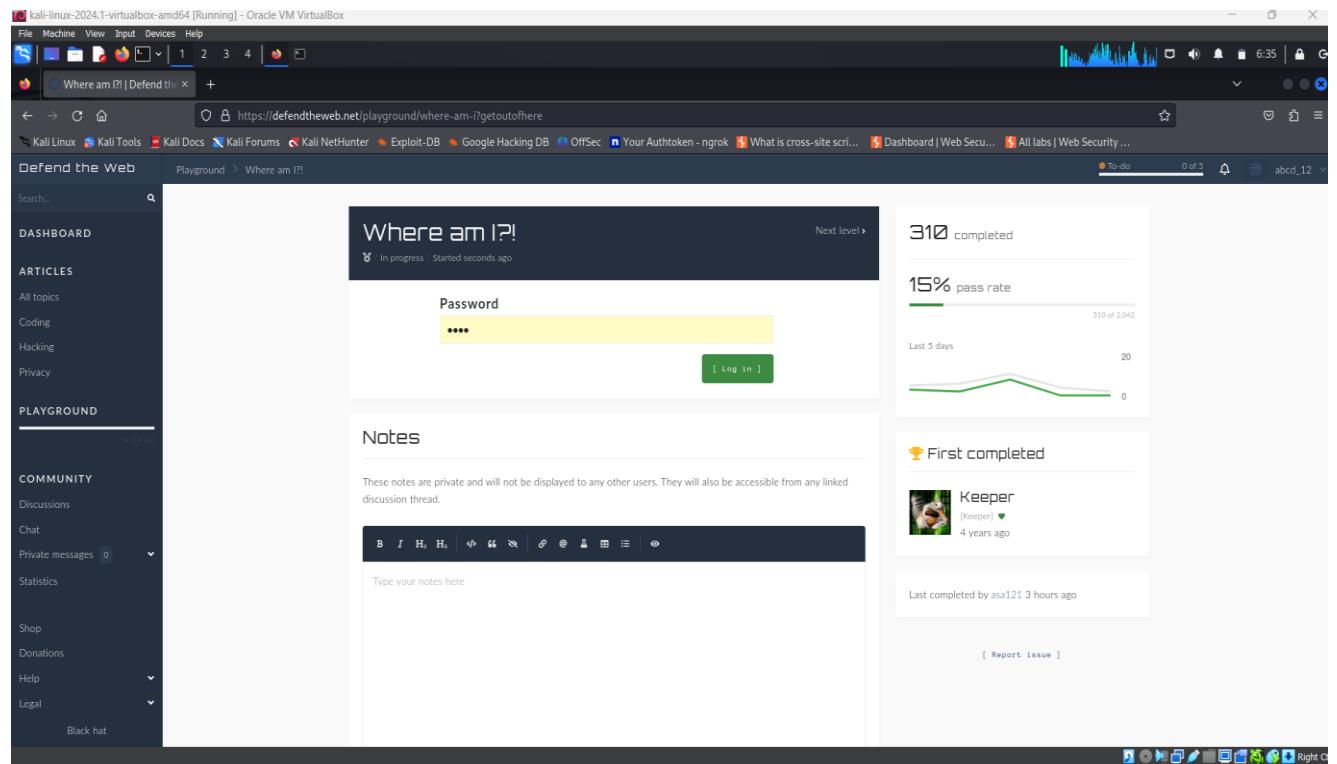
The screenshot shows a browser window with the URL ctflearn.com/challenge/109. The page title is "Challenge - Don't Bump Your Head(er)". The challenge details are: 40 points, Medium difficulty. The description says: "Try to bypass my security measure on this site! <http://165.227.106.113/header.php>". On the right, there's a "Top10" scoreboard and a "Rating" section with a mean rating of 4.60. Below the challenge box is a "Flag" input field containing "CTFLearn{did_this_m3ss_with_yOur_h34d}" and a "Submit" button. The status bar at the bottom shows "15422 solves". The operating system taskbar at the bottom includes icons for various applications like File Explorer, Task Manager, and a weather widget showing 27°C Cloudy.

Step 7 : You can see we're solved the Lab!

The screenshot shows the same browser window after solving the challenge. The challenge box now has a green background and a checkmark icon. The flag input field contains "CTFLearn[h4ck3d]" and the "Submit" button is replaced by a "Solved" button. The status bar at the bottom shows "15423 solves". The "Rating" section now says "Please Rate" and shows a mean rating of 4.60 with five stars filled. The operating system taskbar at the bottom includes icons for various applications like File Explorer, Task Manager, and a weather widget showing 27°C Cloudy.

TASK 3 : <https://defendtheweb.net/playground/where-am-i>

Step 1 : Go to this website



Step 2 : open the burp suite and turn on intercept (first set foxy proxy) write any random password and then go to burp suite and forward the request till you get the request parameter of website.

The screenshot shows a Kali Linux desktop environment. A Firefox browser window is open to the URL <https://defendtheweb.net/playground/where-am-i?getoutofhere>. The page displays a 'Password' field containing '*****'. Below it is a 'Notes' section with the text: "These notes are private and will not be displayed to any other discussion thread." To the right of the browser is the Burp Suite Community Edition v2024.3.1.3 - Temporary Project window. The 'Proxy' tab is selected. In the 'Intercept' tab, a captured POST request is shown:

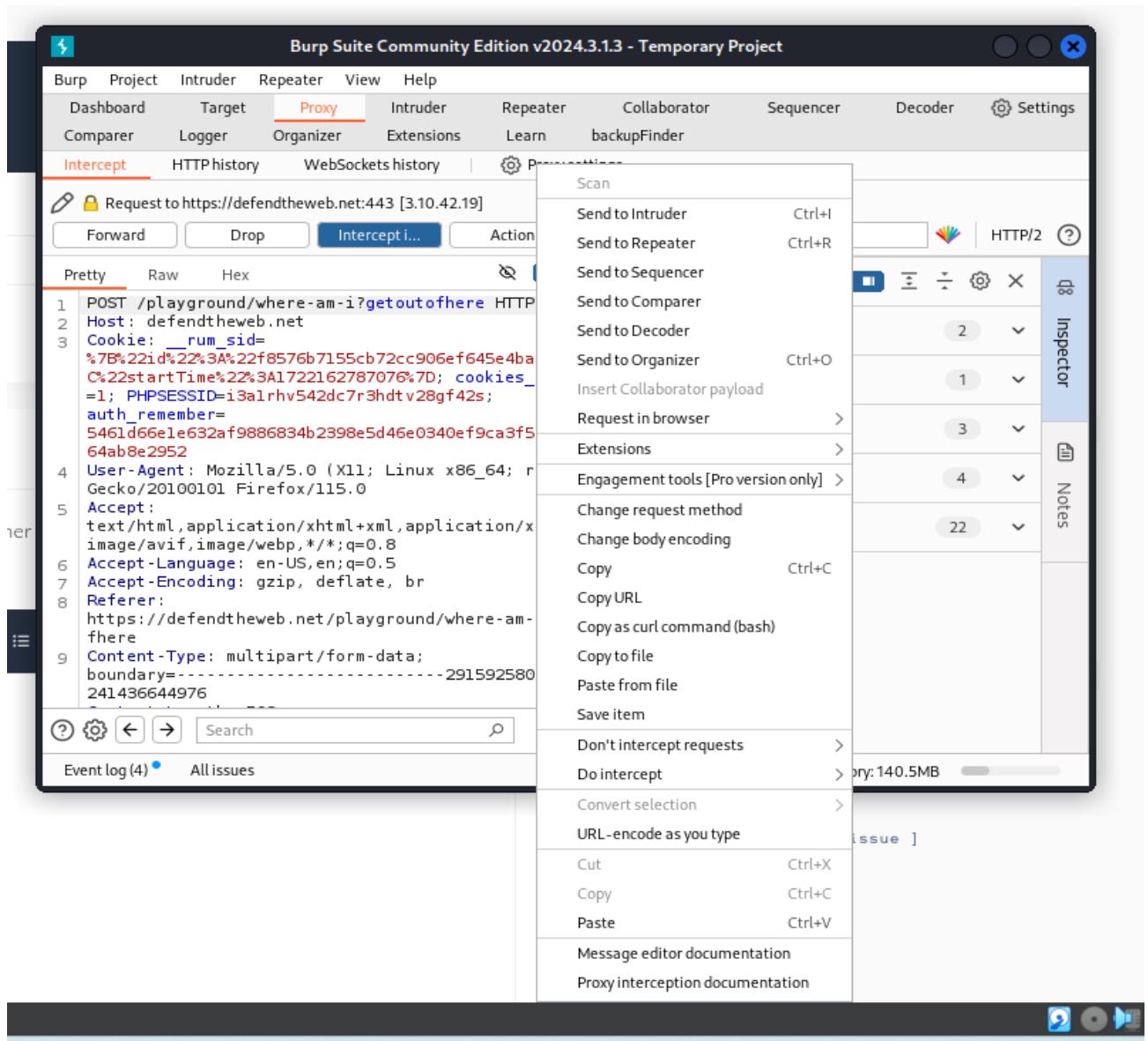
```

1. POST /playground/where-am-i?getoutofhere HTTP/2
2. Host: defendtheweb.net
3. Cookie: run_id=178922d422a3a2f2f857b7155cb72cc906ef645e4b04f34a222
O222PjxwIz23A3l1722162787079670; cookies_dismissed=1
4. PHPSESSID=3a1hv542d3ch3htv28gj42s;
auth_remember=54616661e1652df988883462398b5d46e0340e0fc9ca3f5c0441222
64fb92952
5. User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
6. Accept: text/html,application/xhtml+xml,application/xml;q=0.9,
image/avif,image/webp,*/*;q=0.8
7. Accept-Language: en-US,en;q=0.9
8. Accept-Encoding: gzip, deflate, br
9. Referer: https://defendtheweb.net/playground/where-am-i?getoutofhere
10. Content-Type: multipart/form-data;
boundary=-----201592580830990886
241436644976

```

The Burp Suite interface includes tabs for Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Settings, and a sidebar with sections for Request attributes, Request query parameters, Request body parameters, Request cookies, and Request headers.

Step 3 : click right on the mouse and then click on Send to Repeater .



Step 4 : Go in the Repeater option. And click the send button after that you will able to see pretty option (on middle option (response))

Burp Suite Community Edition v2024.3.1.3 - Temporary Project

Target: https://defendtheweb.net | HTTP/2

Request

```
POST /playground/where-am-i?getoutofhere HTTP/2
Host: defendtheweb.net
Cookie: _rum_sid=7B%22id%22%3A%22f8576b7155cb72cc906ef645e4ba4f34%22%2C%2
2startTIme%22%3A1722162787076%7D; cookies_dismissed=1;
PHPSESSID=i3alrhv542dc7r9hdvt28gf42s; auth_remember=
5461d661e632af9886834b2398e5d46e0340ef9ca3f5c0441422b64a
b8e2952
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0)
Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://defendtheweb.net/playground/where-am-i?getoutofhere
Content-Type: multipart/form-data;
boundary=-----291592580830990886241
436644976
Content-Length: 503
Origin: https://defendtheweb.net
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
```

Response

```
HTTP/2 200 OK
Server: openresty
Date: Sun, 28 Jul 2024 11:04:18 GMT
Content-Type: text/html; charset=UTF-8
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Strict-Transport-Security: max-age=31536000;
includeSubDomains; preload
X-UA-Compatible: IE-Edge
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
Referrer-Policy: no-referrer
X-Xss-Protection: 1
Cache-Control: no-transform
```

Inspector

- Request attributes
- Request query parameters
- Request body parameters
- Request cookies
- Request headers

Notes

Event log (4) All issues

Memory: 140.5MB

After Click the send button :

Burp Suite Community Edition v2024.3.1.3 - Temporary Project

Target: https://defendtheweb.net | HTTP/2

Request

```
POST /playground/where-am-i?getoutofhere HTTP/2
Host: defendtheweb.net
Cookie: _rum_sid=7B%22id%22%3A%22f8576b7155cb72cc906ef645e4ba4f34%22%2C%2
2startTIme%22%3A1722162787076%7D; cookies_dismissed=1;
PHPSESSID=i3alrhv542dc7r9hdvt28gf42s; auth_remember=
5461d661e632af9886834b2398e5d46e0340ef9ca3f5c0441422b64a
b8e2952
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0)
Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://defendtheweb.net/playground/where-am-i?getoutofhere
Content-Type: multipart/form-data;
boundary=-----291592580830990886241
436644976
Content-Length: 503
Origin: https://defendtheweb.net
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
```

Response

```
HTTP/2 200 OK
Server: openresty
Date: Sun, 28 Jul 2024 11:04:18 GMT
Content-Type: text/html; charset=UTF-8
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Strict-Transport-Security: max-age=31536000;
includeSubDomains; preload
X-UA-Compatible: IE-Edge
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
Referrer-Policy: no-referrer
X-Xss-Protection: 1
Cache-Control: no-transform

<!DOCTYPE html>
<html lang="en" class="no-js">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="Content-Type" CONTENT="text/html; charset=UTF-8">
    <title>
      Where am I? | Defend the Web
    </title>
    <meta name="viewport" content="initial-scale=1, maximum-scale=1, user-scalable=1, viewport-fit=cover"
```

Inspector

- Request attributes
- Request query parameters
- Request body parameters
- Request cookies
- Request headers
- Response headers

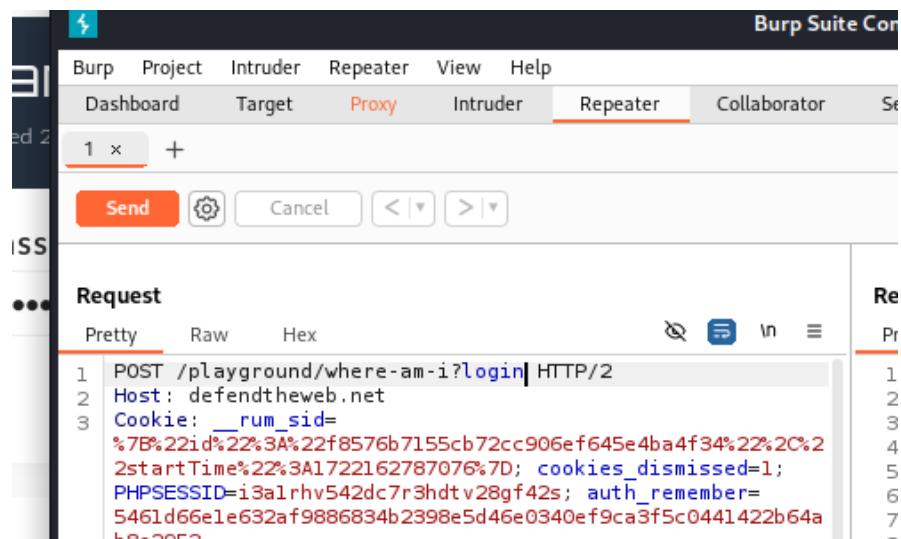
Notes

Event log (4) All issues

30,380 bytes | 1,077 millis

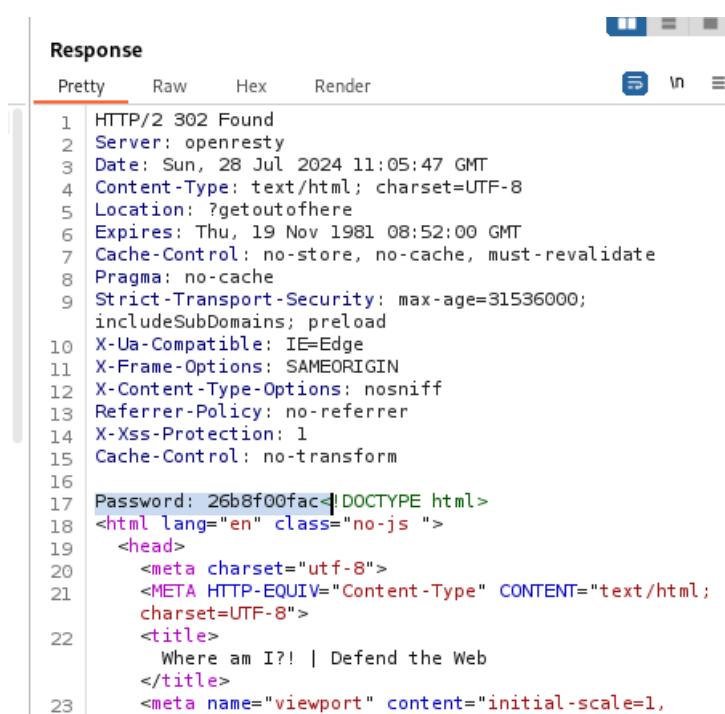
Memory: 154.5MB

Step 5 : you have to change parameter in “login” and then click send button



The screenshot shows the Burp Suite interface with the Repeater tab selected. A POST request is being sent to the endpoint `/playground/where-am-i?login`. The request includes several session cookies such as `_rum_sid`, `PHPSESSID`, and `auth_remember`. The "Pretty" tab is selected in the request pane.

After click on send button :



The screenshot shows the Burp Suite interface with the Response tab selected. The response is a `HTTP/2 302 Found` status with headers including `Server: openresty`, `Date: Sun, 28 Jul 2024 11:05:47 GMT`, and `Content-Type: text/html; charset=UTF-8`. The response body contains a password placeholder `26b8f00fac!` followed by a DOCTYPE declaration and some basic HTML structure. The "Pretty" tab is selected in the response pane.

You can see the password .

Step 6 : now copy it and paste it on the website.

The screenshot shows a web browser window with the following details:

- Address Bar:** https://defendtheweb.net/playground/where-am-i?getoutofhere
- Header:** Shows various links like Docs, Kali Forums, Kali NetHunter, Exploit-DB, Google Hacking DB, OffSec, Your Authtoken - ngrok, What is cross-site scri..., Dashboard | Web S...
- Page Title:** Where am I?!
- Page Content:** A form with a red error message "Invalid login details".
 - Input Field:** Labeled "Password" with the value "*****".
 - Button:** "[Log in]"
- Right Sidebar:** Includes a progress bar (310 col, 15% pa), a chart titled "Last 5 days", and a section titled "First" with a profile picture and the letter "K".

Step 7 : now press the login button.

The screenshot shows a web browser window for the 'Where am I?' challenge on the Defend the Web playground. The main content area displays a green banner with the title 'Where am I?' and a completion message: 'Completed 32 minutes ago' and '32 minutes to complete'. Below the banner is a yellow icon of two hands with blue sparkles above them. The text 'Congratulations' and 'You have completed where am I?!' is displayed. There are two buttons: '[Take on the next challenge]' and '[Share your solution]'. A sidebar on the left contains sections for DASHBOARD, ARTICLES, PLAYGROUND, and COMMUNITY. The PLAYGROUND section is currently selected. The right sidebar shows statistics: '311 completed' (15% pass rate), a chart for 'Last 5 days' showing 20 completions, and a 'First completed' section for 'Keeper' (4 years ago). A note at the bottom encourages sharing accomplishments.

You can see we're solved the lab!

TASK 5 : <https://play.picoctf.org/practice/challenge/262>

Step 1 : Open the given link

The screenshot shows a challenge card for a vulnerability titled "CVE-XXXX-XXXX". The challenge is categorized as "Medium" under "Binary Exploitation" and is part of the "picoCTF 2022" competition. The author is listed as "MUBARAK MIKAIL". A "Hints" button is available, showing one hint has been provided. The challenge description asks for the CVE number in the correct flag format: "picoCTF{CVE-XXXX-XXXX}" where XXXX-XXXX is replaced by the correct numbers. It describes the vulnerability as the first recorded remote code execution (RCE) in 2021 in the Windows Print Spooler Service, which is available across desktop and server versions of Windows operating systems. The service is used to manage printers and print servers. Below the challenge card, it is noted that 17,957 users have solved the challenge, and 56% of users have liked it.

CVE-XXXX-XXXX

Medium Binary Exploitation picoCTF 2022

AUTHOR: MUBARAK MIKAIL

Hints ?

Description

1

Enter the CVE of the vulnerability as the flag with the correct flag format:
picoCTF{CVE-XXXX-XXXX} replacing XXXX-XXXX with the numbers for the matching vulnerability.

The CVE we're looking for is the first recorded remote code execution (RCE) vulnerability in 2021 in the Windows Print Spooler Service, which is available across desktop and server versions of Windows operating systems. The service is used to manage printers and print servers.

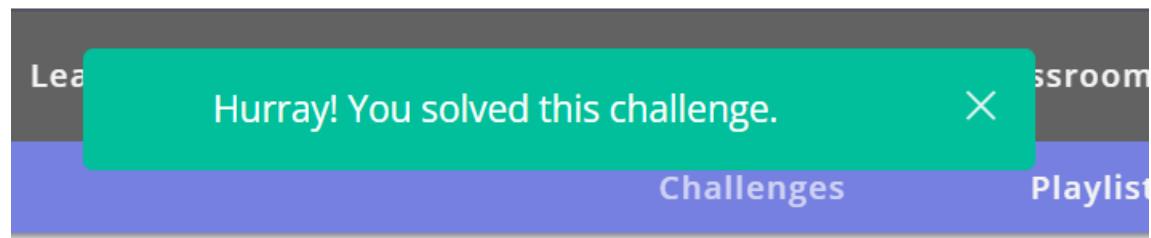
17,957 users solved

56% Liked

Step 2 : You can google it , what's the CVE which we're looking for and you can easily get answer.

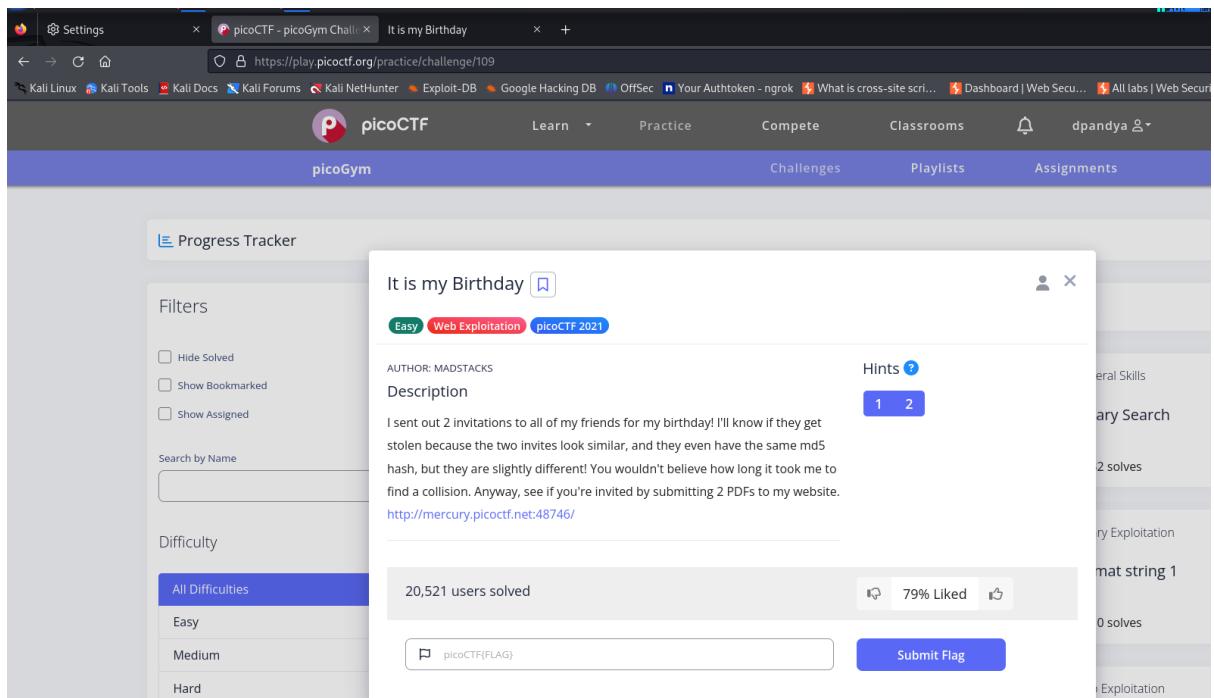
The screenshot shows a challenge page from picoCTF. At the top, there are three tabs: Medium, Binary Exploitation, and picoCTF 2022. Below the tabs, it says AUTHOR: MUBARAK MIKAIL. There is a 'Hints' button with a question mark icon and a blue box containing the number 1. The challenge title is 'Description'. The description text reads: 'Enter the CVE of the vulnerability as the flag with the correct flag format: picoCTF{CVE-XXXX-XXXX} replacing XXXX-XXXX with the numbers for the matching vulnerability. The CVE we're looking for is the first recorded remote code execution (RCE) vulnerability in 2021 in the Windows Print Spooler Service, which is available across desktop and server versions of Windows operating systems. The service is used to manage printers and print servers.' On the left side, there is a 'Filters' section with checkboxes for Hide Solved, Show Bookmarked, and Show Assigned. Below that is a 'Search by Name' input field. Under 'Difficulty', the 'All Difficulties' button is highlighted in blue, while 'Easy' is in grey. In the center, it says '17,955 users solved' with a '56% Liked' button. At the bottom right is a 'Submit Flag' button. The browser's address bar at the top shows the URL play.picoctf.org/practice/challenge/262%20how%20to%20solve%20that.

Step 3 : Click the submit button and you can see We're Solved this lab.

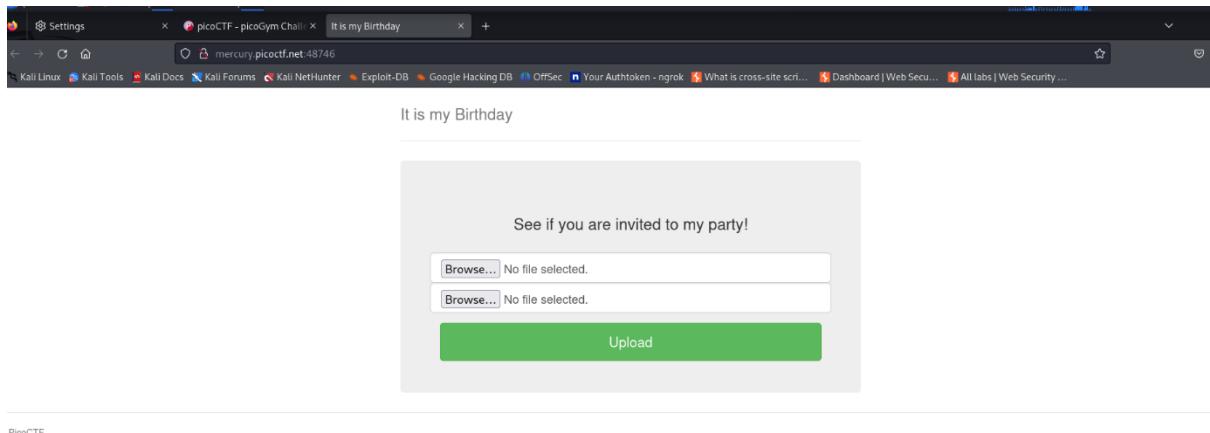


TASK 7 : <https://play.picoctf.org/practice/challenge/109>

Step 1 : Go with this link.

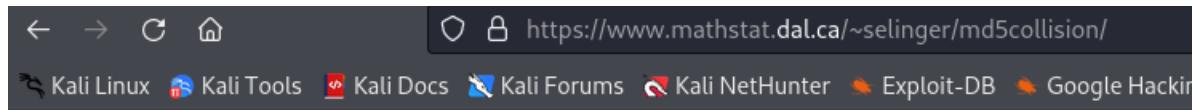


Step 2 : Go to the given URL which is in description.



You have to upload same hash file so that's why I download same hash file on <https://www.mathstat.dal.ca/~selinger/md5collision/> this link.

Step 3 : Click on that site and download both file in Linux version. One file name is 'erase' and another one name is 'hello'. You can see both hash are same.



d131dd02c5e6eec4693d9a0698aff95c2fcab50712467eab4004583eb8fb7f89
55ad340609f4b30283e488832571415a085125e8f7cdc99fd91dbdf280373c5b
d8823e3156348f5bae6dacd436c919c6dd53e2b487da03fd02396306d248cda0
e99f33420f577ee8ce54b67080a80d1ec69821bcb6a8839396f9652b6ff72a70

and

d131dd02c5e6eec4693d9a0698aff95c2fcab50712467eab4004583eb8fb7f89
55ad340609f4b30283e4888325f1415a085125e8f7cdc99fd91dbd7280373c5b
d8823e3156348f5bae6dacd436c919c6dd53e23487da03fd02396306d248cda0
e99f33420f577ee8ce54b67080280d1ec69821bcb6a8839396f965ab6ff72a70

Each of these blocks has MD5 hash 79054025255fb1a26e4bc422aef54eb4. Ben Laurie h Friedle's [Illustrated Guide](#).

Exploits

As we will explain below, the algorithm of Wang and Yu can be used to create files of arb this technique to create pairs of interesting files with identical MD5 hashes:

- Magnus Daum and Stefan Lucks have created [two PostScript files with identical MD5 sums](#).
- Eduardo Diaz has described a [scheme](#) by which two programs could be packed into one file.
- In 2007, Marc Stevens, Arjen K. Lenstra, and Benne de Weger used an improved ve different behaviors. Unlike the old method, where the two files could only differ in a thousand bytes at the end of each file. (Added Jul 27, 2008).
- Didier Stevens used the evilize program (below) to create [two different programs w](#)hich both support MD5. (Added Jan 17, 2009).

An evil pair of executable programs

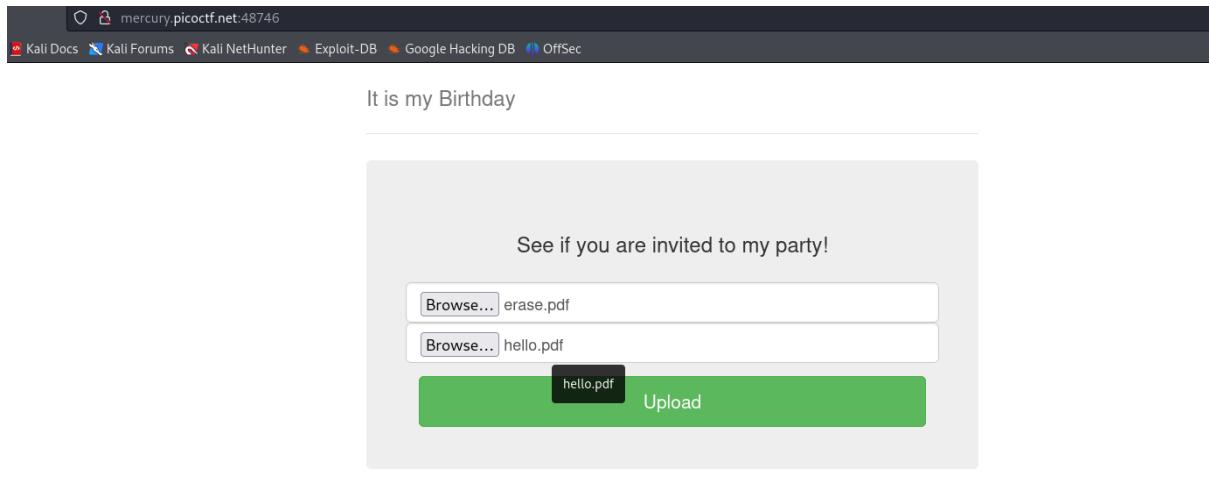
The following is an improvement of Diaz's example, which does not need a special extraction tool.

- **Windows version:**
 - [hello.exe](#). MD5 Sum: cdc47d670159eef60916ca03a9d4a007
 - [erase.exe](#). MD5 Sum: cdc47d670159eef60916ca03a9d4a007
- **Linux version (i386):**
 - [hello](#). MD5 Sum: da5c61e1edc0f18337e46418e48c1290
 - [erase](#). MD5 Sum: da5c61e1edc0f18337e46418e48c1290

Step 4 : Now copy the files on your folder for change the format (we're converting into pdf format) .

```
File Hunter Exploit-DB Google Hacking DB
kali@kali: ~/PicoCTF/challenge_109
503cb8f7f09
ben@kali:~/Downloads$ cp /home/kali/Downloads/erase erase.pdf
ben@kali:~/Downloads$ cp /home/kali/Downloads/hello hello.pdf
ben@kali:~/Downloads$ 
265fb1a26e4bc422aef54eb4. Ben Laurie has a nice website that visualizes this MD5 collision. For a non-technic
```

Step 5 : now upload the both files on that website.



Step 6 : Now click the upload button. And you can see the flag which we're needed. Now copy the flag.

```

← → ⌂ mercury.picoctf.net:48746/index.php
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
<?php
if (isset($_POST["submit"])) {
    $type1 = $_FILES["file1"]["type"];
    $type2 = $_FILES["file2"]["type"];
    $size1 = $_FILES["file1"]["size"];
    $size2 = $_FILES["file2"]["size"];
    $SIZE_LIMIT = 1024;
    if ($size1 < $SIZE_LIMIT && $size2 < $SIZE_LIMIT) {
        if ((($type1 == "application/pdf") && ($type2 == "application/pdf")) && ($contents1 == file_get_contents($_FILES["file1"]["tmp_name"]))) {
            $contents2 = file_get_contents($_FILES["file2"]["tmp_name"]);
            if ($contents1 != $contents2) {
                if (md5_file($_FILES["file1"]["tmp_name"]) == md5_file($_FILES["file2"]["tmp_name"])) {
                    echo "MD5 hashes do not match!";
                    die();
                } else {
                    echo "MD5 hashes are not different!";
                    die();
                }
            } else {
                echo "Files are not different!";
                die();
            }
        } else {
            echo "Not a PDF!";
            die();
        }
    } else {
        echo "File too large!";
        die();
    }
}
// FLAG: picoCTF{c0ngr4ts_u_r_Inv1t3d_aebcbf39}

?>
<!DOCTYPE html>
<html lang="en">
<head>
<title>It is my Birthday</title>
<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css" rel="stylesheet">
<link href="https://getbootstrap.com/docs/3.3/examples/jumbotron-narrow/jumbotron-narrow.css" rel="stylesheet">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

```

Step 7 : paste it on main website.

It is my Birthday

Easy Web Exploitation picoCTF 2021

AUTHOR: MADSTACKS

Description

I sent out 2 invitations to all of my friends for my birthday! I'll know if they get stolen because the two invites look similar, and they even have the same md5 hash, but they are slightly different! You wouldn't believe how long it took me to find a collision. Anyway, see if you're invited by submitting 2 PDFs to my website.

<http://mercury.picoctf.net:48746/>

20,525 users solved

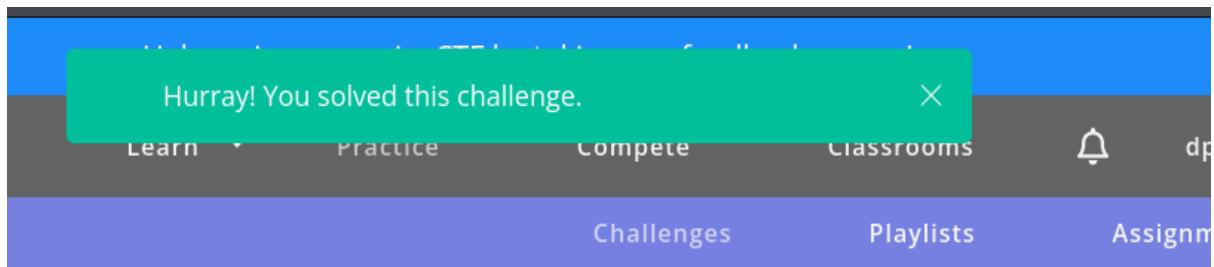
Hints ?

1 2

picoCTF{c0ngr4ts_u_r_1nv1t3d_aebcbf39}

Submit Flag

Step 8 : Press the submit flag button.



You can see we're solved this challenge.

TASK 8 : <https://play.picoctf.org/practice/challenge/4>

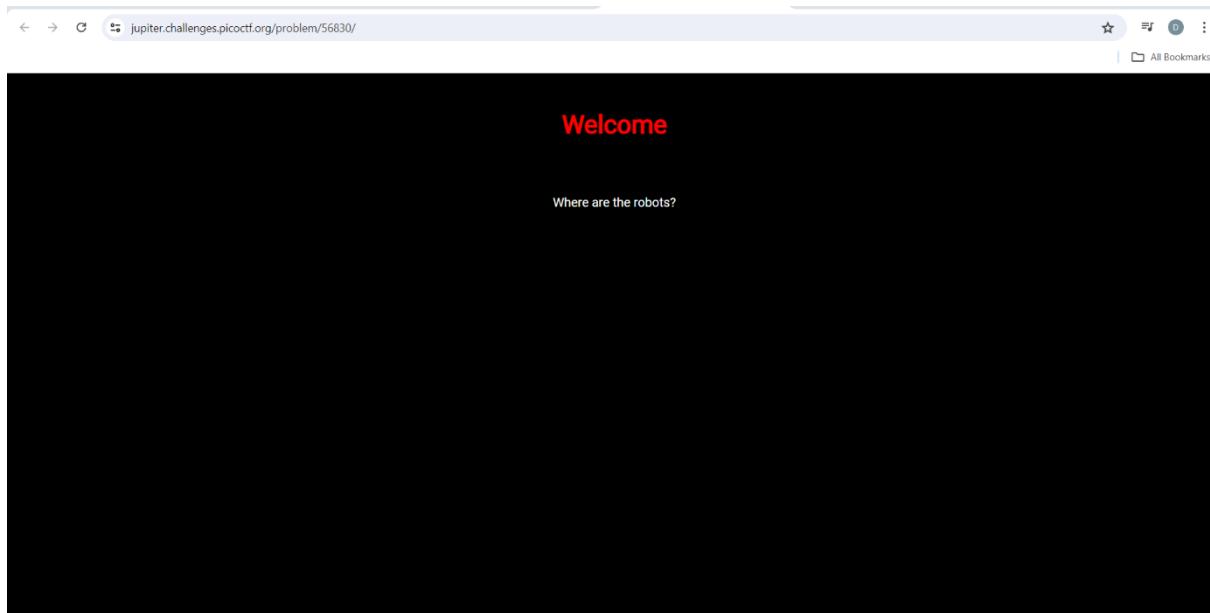
Step 1 : Go with this link.

The screenshot shows a challenge card for 'where are the robots'. The challenge is categorized under 'Easy', 'Web Exploitation', and 'picoCTF 2019'. It was authored by ZARATEC/DANNY. The description asks if you can find the robots and provides a link: <https://jupiter.challenges.picoctf.org/problem/56830/>. Below the description, it says 75,170 users solved the challenge. A 'Submit Flag' button is visible at the bottom right. On the right side of the card, there's a sidebar showing user statistics: 89% Liked, 84%, and 95%. The background of the page shows a sidebar with filters like 'Progress Track', 'Filters', and 'Difficulty' (set to 'All Difficulties'). The top navigation bar includes 'Learn', 'Practice', 'Compete', 'Classrooms', and a user profile for 'dpandya'. The address bar at the top of the browser window shows the URL: play.picoctf.org/practice/challenge/4.

Step 2 : click on the link

<https://jupiter.challenges.picoctf.org/problem/56830/>

And you can see the page look like this.



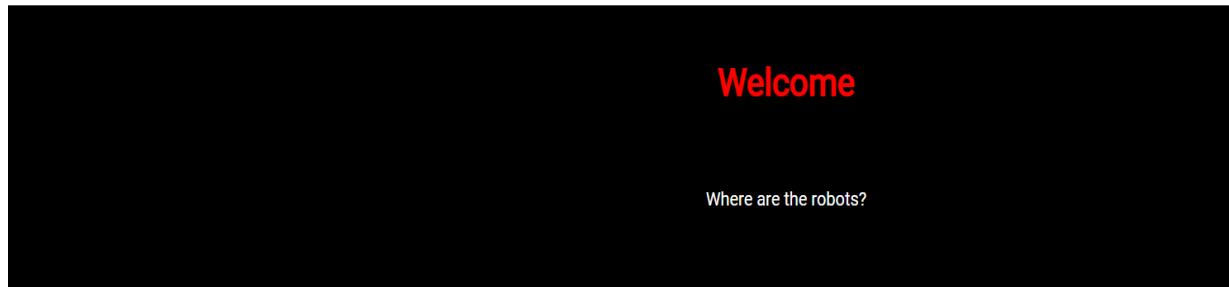
You can see in view page resources ,there is nothing.

A screenshot of a browser window displaying the page source code. The address bar shows the URL: view-source:https://jupiter.challenges.picoctf.org/problem/56830/. The source code is as follows:

```
Line wrap □
1 <!doctype html>
2 <html>
3   <head>
4     <title>Welcome</title>
5     <link href="https://fonts.googleapis.com/css?family=Monoton|Roboto" rel="stylesheet">
6     <link rel="stylesheet" type="text/css" href="style.css">
7   </head>
8
9   <body>
10    <div class="container">
11      <header>
12        <h1>Welcome</h1>
13      </header>
14      <div class="content">
15        <p>Where are the robots?</p>
16      </div>
17      <footer></footer>
18    </div>
19  </body>
20 </html>
```

Step 3 : write robots.txt in parameter in search bar.

← → ⌛ https://jupiter.challenges.picoctf.org/problem/56830/robots.txt



And press enter.

← → ⌛ https://jupiter.challenges.picoctf.org/problem/56830/robots.txt

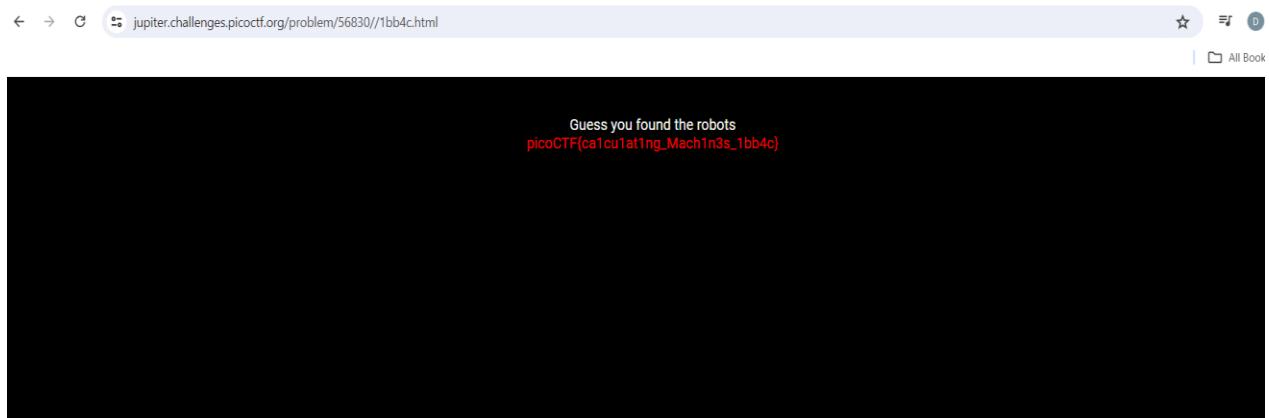
```
User-agent: *
Disallow: /1bb4c.html
```

Step 4 : now write with given file name /1bb4c.html in the parameter.

← → ⌛ https://jupiter.challenges.picoctf.org/problem/56830//1bb4c.html

```
User-agent: *
Disallow: /1bb4c.html
```

Now press enter.

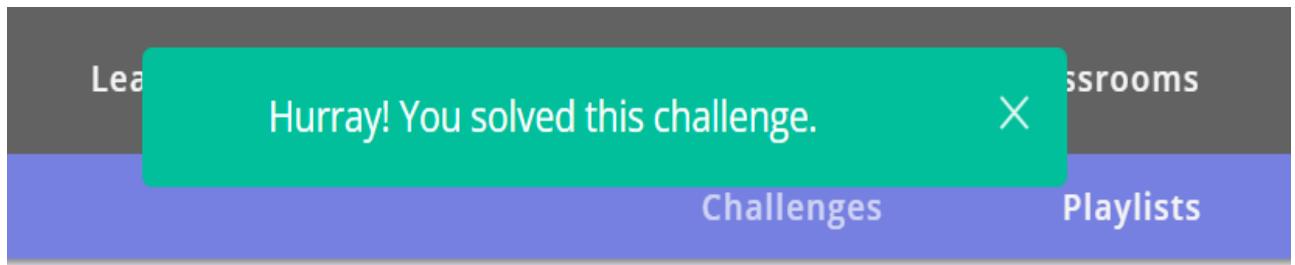


You can we're capture the flag.

Step 5 : Now copy the flag and paste it on website.

A screenshot of a challenge card from picoCTF. The title is 'where are the robots' with a bookmark icon. It has difficulty levels 'Easy' and 'Web Exploitation' and was created in 'picoCTF 2019'. The author is 'ZARATEC/DANNY'. There is a 'Description' section with the text: 'Can you find the robots?' and a link 'https://jupiter.challenges.picoctf.org/problem/56830/' (link). Below the description, it says 'or http://jupiter.challenges.picoctf.org:56830'. A progress bar shows '75,170 users solved' and '89%' liked. A blue button at the bottom right says 'Submit Flag'. The bottom of the screen shows a Windows taskbar with various icons.

& click the submit flag button.



You can see we're solved this challenge.

THANK YOU

CFSS PAID Internship Program

Theory Questions:

QUE : 1. Explain the difference between vulnerability assessment and penetration testing.

Ans :

1. Penetration Testing: Designed for critical real-time systems.

Vulnerability Assessments: Geared towards non-critical systems.

2. Penetration Testing: Best suited for physical environments and network architectures.

Vulnerability Assessments: Ideal for lab environments.

3. Penetration Testing: Non-intrusive, focusing on documentation and environmental review and analysis.

Vulnerability Assessments: Involves a comprehensive analysis and thorough review of the target system and its environment.

4. **Penetration Testing:** Cleans up the system and provides a final report.
Vulnerability Assessments: Attempts to mitigate or eliminate potential vulnerabilities of valuable resources.
5. **Penetration Testing:** Gathers targeted information and inspects the system.
Vulnerability Assessments: Assigns quantifiable value and significance to available resources.
6. **Penetration Testing:** Tests sensitive data collection.
Vulnerability Assessments: Discovers potential threats to each resource.
7. **Penetration Testing:** Determines the scope of an attack.
Vulnerability Assessments: Creates a directory of assets and resources in a given system.
8. **Penetration Testing:** Focuses on uncovering unknown and exploitable weaknesses in standard business processes.
Vulnerability Assessments: Lists known software vulnerabilities that could be exploited.
9. **Penetration Testing:** A simulated cyberattack conducted by experienced ethical hackers in a controlled environment.

Vulnerability Assessments: An automated assessment performed using automated tools.

10. **Penetration Testing:** A goal-oriented procedure conducted in a controlled manner.
Vulnerability Assessments: A cost-effective and often safe assessment method.

 11. **Penetration Testing:** Identifies only the exploitable security vulnerabilities.
Vulnerability Assessments: Identifies, categorizes, and quantifies security vulnerabilities.
-

QUE : 2. Describe the role of social engineering in a penetration test and how it can be mitigated.

Ans :

Penetration tests simulate the actions of hackers to evaluate your defences and uncover vulnerabilities, forming a vital component of any robust cybersecurity program. In particular, a social engineering penetration test involves collaborating with ethical hackers who replicate common social engineering attacks to assess your defences.

These actions may include:

- Using social engineering tactics to infect a user's computer.
- Distributing phishing emails designed to deceive users into divulging confidential information.
- Employing voice phishing schemes that trick users into disclosing sensitive data, such as usernames and passwords.

During a penetration test, the ethical hackers emulate the strategies a cybercriminal would use to exploit the human element of security. This approach is crucial for gauging how vulnerable your employees are to social engineering attacks. Unlike other cyberattacks that focus on technological weaknesses, social engineering targets the awareness and vigilance of your staff.

QUE : 3. What is privilege escalation, and how is it achieved during a penetration test?

Ans :

Privilege escalation is the process of obtaining elevated access to resources typically protected from an application or user. This involves exploiting

vulnerabilities to gain higher levels of access than initially granted, such as acquiring administrative or root access on a system.

Types of Privilege Escalation

- **Horizontal Privilege Escalation:** Occurs when a user acquires the privileges of another user with a similar access level.
- **Vertical Privilege Escalation:** Happens when a user gains higher privileges, such as administrative or root access.

How Privilege Escalation is Achieved During a Penetration Test

Identifying Vulnerabilities:

- **Unpatched Software:** Exploiting known vulnerabilities in outdated software.
- **Misconfigurations:** Leveraging system misconfigurations or weak security settings.
- **Weak Passwords:** Cracking weak or default passwords to escalate privileges.

Exploiting Vulnerabilities:

- **Local Exploits:** Using vulnerabilities accessible by an authenticated user to gain higher privileges.

- **Remote Exploits:** Exploiting vulnerabilities remotely to elevate privileges on the target system.

Common Techniques:

- **Buffer Overflows:** Overwriting memory to execute arbitrary code with elevated privileges.
- **Kernel Exploits:** Targeting vulnerabilities in the operating system kernel.
- **DLL Hijacking:** Inserting malicious Dynamic Link Libraries (DLLs) to be loaded by legitimate applications.
- **SUID/Sgid Binaries:** Exploiting set-user-ID (SUID) or set-group-ID (SGID) binaries that run with elevated privileges.
- **Password Hash Dumping:** Extracting and cracking password hashes.
- **Insecure File Permissions:** Modifying or replacing files that are incorrectly accessible.

Example Scenario of Vertical Privilege Escalation

1. **Initial Access:** The penetration tester gains access to a system with limited user privileges.

2. **Information Gathering:** Identifies installed software, running services, and system configurations for potential vulnerabilities.
3. **Exploit Development:** Develops or uses an existing exploit targeting a specific vulnerability.
4. **Exploit Execution:** Executes the exploit to gain higher privileges, such as administrative access.
5. **Post-Exploitation:** Performs actions requiring elevated privileges, like installing backdoors, extracting sensitive data, or further escalating privileges.

Mitigation Measures

- **Regular Patching and Updates:** Keep software and systems updated to prevent exploitation of known vulnerabilities.
- **Strong Password Policies:** Enforce strong, unique passwords and use multi-factor authentication.
- **Least Privilege Principle:** Grant users the minimum level of access necessary to perform their tasks.
- **Security Audits and Monitoring:** Regularly audit systems for vulnerabilities and monitor for suspicious activities.

- **Configuration Management:** Ensure systems are configured securely, following best practices.

Implementing these measures can significantly reduce the risk of successful privilege escalation attacks.

QUE – 4 : Discuss the significance of a honeypot in a cybersecurity environment.

Ans :

A honeypot is a bait-involving system or network feature fabricated to lure cybercrooks and probe into their activities. It functions as an eye at a distance, and an early-warning detector, helping real-life systems to be more secure, by taking attackers, and collecting valuable information about their methods.

Significance of a Honeypot in a Cybersecurity Environment

Threat Detection : Early Warning System: Honeypots can detect new and emerging threats that traditional security measures might miss. Through attracting intruders, they offer the first indications of possible security breakages. **Attack Patterns:** They enable the

backstory behind the tools, techniques, and procedures (TTPs) carried out by the attackers, thus enabling the organizations to be prepared and respond to the upcoming attacks.

Early Warning System: Honeypots can detect new and emerging threats that traditional security measures might miss. Through attracting intruders, they offer the first indications of possible security breakages.

Attack Patterns : They help in understanding the tactics, techniques, and procedures (TTPs) used to carry out an attack, therefore, the organizations can anticipate and safeguard against the attacks.

Incident Response : Behaviour Analysis: Honeypots allow security teams to observe attackers in a controlled environment, which helps in developing better incident response strategies . Forensics: They provide a wide range of forensic data that can be analysed to comprehend the effectiveness and origins of the attacks, thus, post-incident investigations will be facilitated.

Behaviour Analysis: Honeypots are used by security teams' to observe attackers in a controlled environment, which helps in developing better incident response strategies.

Forensics: Honeypots give a lot of data that can be analysed to understand the extent and nature of attacks, will be used in the post-incident investigation.

Vulnerability Assessment : Identifying Weaknesses: Through observation of how attackers interact with a honeypot, security teams can spot vulnerabilities and weaknesses in their systems that are actual ones.

Testing Defensive Measures: Honeypots used for the purpose of testing and thus, validating the Effy

Identifying Weaknesses: Observing the activities of attackers related to the honeypot allows security teams to pinpoint vulnerabilities and weaknesses in their systems.

Testing Defensive Measures: Honeypots can be deployed to test different security frameworks and thus to validate the ongoing schemes and controls that are doing well in place.

Deception and Diversion : Attackers Wasteful Use of Resources: Honeypots waste the attacker's time and resources, redirecting their attention away from actual weaknesses and the risk of their attacks being discovered. **Removing False Positives:** The set of honeypots helps prevent false alarms of the security system by distinguishing genuine attacks from benign activities.

Resource Wastage for Attackers: Honeypots waste the time and resources of attackers by attracting them to the decoy machines and making it more difficult for them to succeed.

Types of Honeypots

Production Honeypots: They are set up in the organization's production environment to spot and redirect real-time attacks.

Research Honeypots: Mainly used by researchers to analyse the methods of attack, as well as, to detect the hackers' behaviour by gathering information.

Implementation Considerations

Placement: It is imperative to place the honeypots properly so that they may attract the attackers but not create vulnerability in the security of the actual systems.

Legal and Ethical Issues: Companies must be mindful of the legal and ethical problems associated with the employment of the honeypots, especially with regard to privacy and data protection.

Resource Allocation: Setup and runtime for honeypots are resource and expertise intensive activities and they may need to be manned periodically.

QUE 5 : Explain the concept of "zero-day exploit" and discuss the ethical considerations when encountering such vulnerabilities during penetration testing.

Ans :

Zero-Day Exploit

A zero-day exploit is a type of cyber-attack that avails of a software flaw that is not identified by the software vendor or the public. As the vendor does not have information on the vulnerability, there are no patches available, thus the zero-day attack may become exceedingly dangerous. These are termed zero-day exploits since developers have zero days to correct the error before the user is exploited.

Key Characteristics of Zero-Day Exploits

Unknown Vulnerability: The vulnerability is not known to the vendor or the public. Thus, it is really hard to defend oneself.

No Available Patch: There is no existing fix or patch for the vulnerability at the time of the exploit.

High Impact: Zero-day exploits can cause significant damage, as they can bypass traditional security measures.

Exclusivity: These are typically private or not bought as tools and programs on the black market by the technology companies. They are the exclusive property of either the intelligence or terrorists, making them a rare commodity and undercover to the authorities.

Ethical Considerations During Penetration Testing

Responsible Disclosure : Vendor Notification: The ethical hackers who use penetration testing in their work should immediately inform the impacted vendor about a zero-day vulnerability so that they can develop a patch or fix.

Coordinated Disclosure: Collaboration with the vendor in coordinating the public disclosure of the vulnerability in a manner that reduces the risk to users.

Vendor Notification: Upon discovering a zero-day vulnerability, ethical penetration testers should promptly notify the affected vendor to allow them to develop a patch or fix.

Coordinated Disclosure: With the vendor to organize the public exposure of the vulnerability with the aim of safeguarding users' safety.

Non-Disclosure Agreements (NDAs):Client Confidentiality: Penetration testers frequently work with NDAs that explicitly demand them to maintain secrecy of certain vital details of their work, such as new-day vulnerabilities. Balancing Disclosure and **Confidentiality:** On the one hand, testers have to protect the client's interests through confidentiality. On the other hand, they still have to report such vulnerabilities to scale the overall ethical responsibility.

Client Confidentiality: Penetration testers often work under NDAs that require them to keep findings confidential, including zero-day vulnerabilities.

Balancing Disclosure and Confidentiality: Testers must balance the need to protect client confidentiality with the broader ethical responsibility to disclose vulnerabilities to vendors.

Usage of Zero-Day Exploits : Avoiding Harm: By using ethical means, penetration testers can avoid the

application of zero-day exploits, which may lead to harm or disturbance of a client's system. **Informed Consent:** When clients are informed about the potential risks involved in using zero-day exploits they take part in the testing by giving their explicit permission.

Avoiding Harm: Ethical penetration testers should avoid using zero-day exploits that could cause harm or disrupt client systems.

Informed Consent: Clients should be informed about the potential risks associated with using zero-day exploits during testing, and their explicit consent should be obtained.

Legal Implications : Compliance with Laws: Testers must disregard the criminalization of actions that are cyber law related. The statutes in different jurisdictions, including the hacker state law, should not be trivialized and evaluated in the right way.

Compliance with Laws: Testers are the leader of the pack, and thus they must be on the pinpoint of the different laws, national, and international dealing with cybersecurity and data protection.

Avoiding Illegal Activities: One crime that could be verified is when hackers sell or trade zero-day attacks. Beyond that, breaches may result in prison or a fine.

QUE – 6 : Explain the concept of "full disclosure" versus "responsible disclosure" in vulnerability disclosure processes. How do ethical hackers navigate these principles during penetration testing engagements?

Ans :

The disclosure includes both Full Disclosure and Responsible Disclosure, which are two methods of revealing a vulnerability.

Full Disclosure

Full Disclosure happens in cases when the publicly release of a detailed report of a vulnerability, including ways through which it may be exploited, is done, without necessarily notifying the affected vendor first. The main purpose of the disclosure is to inform the public and force the system or software vendor to make the security fix as soon as possible to break the vulnerabilities.

Pros:

Rapid Response: Compels producers to react to errors more quickly due to public pressure, thereby reducing the window of exposure.

Public Awareness: The provision and dissemination of warnings to consumers and other stakeholders (like investors and service providers) so that they can take preventive measures will make clear to all the people involved how urgent the issue is.

Cons:

The danger that the vulnerability is exploited: During the time between the discovery of the vulnerability and the availability of a patch, it is entirely possible for hackers to attack the vulnerability of the system thereby affecting many people.

Limited Coordination: It is usually the vendor who seeks a way to request the problem to be fixed and so it is not unusual that there is some delay or that the fix is not final.

Responsible Disclosure :

Responsible Disclosure is the safer approach because the reporter ideally should be disclosing the vulnerabilities to the vendor privately, then the vendor

would then have time to design and distribute a patch to the users before publicly releasing the details. This process usually includes a discussion of the period during which the disclosure is made.

Pros:

Reduced Risk: It minimizes the threat posed by the vulnerability to attackers for a short duration until the patch is released.

Vendor Collaboration: Manufacturers create holistic and effective patches through this system.

Cons:

Delayed Awareness: It is likely that the users and other stakeholders will remain unaware of the vulnerability until the release of a patch, which in some cases may be the only protection left from such an attack.

Steps for Ethical Hackers :

Initial Assessment: Try to gauge the severity as the level of the risk and the potential effect of the threat. Assess (if applicable) the degree to which the

vulnerability can affect the client only as opposed to the broadness of the threat.

Try to gauge the severity as the level of the risk and the potential effect of the threat.

Assess (if applicable) the degree to which the vulnerability can affect the client only as opposed to the broadness of the threat.

Client Communication: Send a message to the client to let them know about the weakness and discuss with them the possible impact and risk. Get the client to concede the way to go, especially the communication with the vendor.

Send a message to the client to let them know about the weakness and discuss with them the possible impact and risk.

Get the client to concede the way to go, especially the communication with the vendor.

Vendor Notification: Ethical hackers shall use the first option, i.e., the alert, to get the client to understand the relevance of security -- i.e., the Findings. Give a detailed reading of the bug, like how to recreate it and its solutions.

If the vulnerability is severe and impacts the vendor's product, ethical hackers should notify the vendor directly.

Give all the details of the vulnerability, demonstrate it through the proof of concept and suggest its repairs.

Coordinated Disclosure: Be with the vendor to have them agree on the schedule for publishing the information so that there is enough time for the fixing and testing of the patch. Coordinate every party – client, and vendor – to the extent where all are aware and ready for disclosure.

Engage actively with a vendor to fix a date for the public revelation for a time when there is enough time for the development of the patch and its evaluation.

Facilitate the process of information exchange between the client and the vendor to be prepared and informed workaday.

Public Disclosure : After the period that has elapsed, publish the security hole data, except with regards to the description of the flaw, what the affected systems are, and how to fix it. Ensure that the disclosure is clear and informative, helping other stakeholders to understand the risks and necessary actions.

After the period, let them know you will make the vulnerability public, give them all the information about the flaw, what devices are impacted and the ways of resolving the issue.

See to it that the disclosure is clear and informative, thus giving other stakeholders clear insight into the risks and needed actions.

Ethical Considerations : Begin with the user's safety and digital security in mind by reducing the functionality used by attackers. Be honest and transparent in all your discussions. Honor all the laws and contracts including NDA's if they are present.

Ensure all the users protect their rights and liberty by making a reduction in the risk of exploitation their main goal.

Democratize information and communications, allowing all sides to communicate about all issues related to their jobs.

Rights beyond legal and contractual rolls are not to be violated in any manner, including non-disclosure agreements.

QUE – 7 : Discuss the ethical considerations of using automated tools and scripts in penetration testing. How can ethical hackers ensure the responsible use of automation while maintaining accuracy and thoroughness in their assessments?

Ans :

Ethical Considerations of Using Automated Tools and Scripts in Penetration Testing

1. Scope and Permission

- Authorization: Use automated tools strictly within the scope of authorized penetration testing. Unauthorized use can lead to serious legal and ethical issues.
- Scope Adherence: Configure tools to operate within the defined scope to prevent unintended disruptions or attacks on other systems.

2. Accuracy and Reliability

- False Positives/Negatives: Automated tools may produce false positives or negatives. Ethical hackers must manually verify findings

to ensure accuracy and avoid misleading results.

- Tool Limitations: Understand what your tools can and cannot do. Sole reliance on automation might overlook nuances that require a manual touch and expert analysis.

3. Impact on Systems

- Resource Consumption: Automated scanning can heavily load target systems, causing performance issues or outages. Calibrate tools to minimize impact and avoid harm.
- Testing in Production Environments: Be cautious with live systems. When possible, conduct tests in a staging environment to avoid disruptions.

4. Data Privacy and Confidentiality

- Handling Sensitive Data: Automated tools might collect sensitive information during scans. Ensure such data is securely handled and not exposed or misused.
- Data Protection: Follow best practices for data protection and confidentiality, including secure storage and handling of results.

5. Communication and Reporting

- Clear Reporting: Document and communicate results clearly, noting any limitations or potential inaccuracies. Ensure clients understand the context of findings.
- Transparency: Be transparent about using automated tools. Explain their role and how they complement manual testing efforts.

Ensuring Responsible Use of Automation

1. Complement Manual Testing

- Hybrid Approach: Use automated tools to complement, not replace, manual testing. Automation handles repetitive tasks, while manual testing addresses complex issues.
- Validation: Cross-check automated results with manual validation for thoroughness and accuracy.

2. Customization and Configuration

- Tailor Tools: Customize and configure tools to fit the specific engagement requirements. This targets the correct scope and reduces false positives.
- Regular Updates: Keep tools updated to ensure they're effective against new vulnerabilities and threats.

3. Impact Mitigation

- Controlled Testing: Run automated scans during off-peak hours or in controlled environments to minimize production system impacts.
- Rate Limiting: Use rate limiting and safeguards to prevent system overloads during scans.

4. Ethical Guidelines and Standards

- Follow Best Practices: Adhere to industry best practices and ethical guidelines for using automated tools, respecting legal constraints and maintaining professional conduct.
- Continuous Learning: Stay updated on the latest developments in automated tools and security practices to ensure responsible and effective use.

Balancing automation efficiency with careful oversight and validation enables ethical hackers to enhance penetration testing while upholding high standards of accuracy and ethical behaviour.

QUE – 8 : Explain the concept of "bug bounty hunting" from the perspective of a gray hat hacker. How does a gray hat hacker decide which vulnerabilities to report and which to exploit for personal gain?

Ans :

Bug Bounty Hunting from a Gray Hat Hacker's Perspective :

1. Definition of a Gray Hat Hacker

A gray hat hacker operates in the space between ethical (white hat) and unethical (black hat) hacking. These individuals may uncover vulnerabilities without harmful intent but often seek personal benefit, whether through selling the information or gaining recognition.

2. Bug Bounty Hunting Overview

Organizations offer bug bounty programs to motivate security researchers to find and disclose vulnerabilities. These programs reward researchers monetarily or through other incentives for responsibly reporting bugs.

3. Decision-Making in Reporting vs. Exploiting

a. Ethical and Legal Considerations Gray hat hackers often engage with bug bounty programs, reporting vulnerabilities through official channels. This approach aligns with program terms and aids in enhancing security. However, when these hackers choose to exploit vulnerabilities for personal gain—such as selling the information or using it for unauthorized access—they cross into unethical territory, breaching legal and ethical norms.

b. Criteria for Decision-Making

- **Severity and Impact:** Gray hat hackers assess the severity and potential impact of a vulnerability. They are more likely to report critical vulnerabilities to maximize rewards or gain public recognition.
- **Program Rules and Rewards:** The rules and rewards of a bug bounty program play a crucial role. If the incentives are seen as insufficient, some hackers may be tempted to exploit the vulnerability instead.
- **Ethical Boundaries:** While gray hat hackers might follow a personal code of ethics, their actions can vary. Some choose to report vulnerabilities responsibly, whereas others might exploit them if

they find the rewards inadequate or believe they can avoid consequences.

- Reputation and Recognition: The desire for recognition within the cybersecurity community drives some gray hat hackers. Reporting significant vulnerabilities can enhance their reputation, whereas exploitation might offer short-term benefits but harm their long-term credibility.

4. Navigating Ethical Boundaries

Gray hat hackers must consider the ethical implications of their actions and aim to balance personal gain with responsibility. Participating in bug bounty programs and adhering to responsible disclosure practices can help maintain ethical standards. Exploiting vulnerabilities without authorization poses legal risks, including potential criminal charges, which must be weighed against any perceived benefits.

5. Impact on the Security Landscape

Gray hat hackers contribute positively to cybersecurity by reporting vulnerabilities, helping organizations address security issues and protect users. Conversely, exploiting vulnerabilities for personal gain erodes trust and can cause harm to organizations, users, and the broader security community.

QUE - 9. Explain the concept of "hacktivism" and discuss the ethical considerations involved in using hacking techniques to promote social or political causes. How does a gray hat hacker justify their actions in the name of activism?

Ans :

Hacktivism is a blend of "hacking" and "activism" and refers to the use of hacking techniques to promote social or political causes. Hacktivists leverage their technical skills to disrupt systems, leak information, or deface websites, aiming to draw attention to their cause or to protest against policies or practices they consider unjust.

Ethical Considerations in Hacktivism

1. Intent and Purpose: The primary ethical consideration is the intent behind the hacktivist activities. While hacktivists may believe they are

fighting for a just cause, their actions can still cause harm, disrupt services, or violate laws.

2. Consent and Harm: Hacktivism often involves actions taken without the consent of those affected, potentially causing harm to innocent third parties. For example, a Distributed Denial of Service (DDoS) attack on a government website might also affect citizens relying on that website for essential services.

3. Legality: Hacktivism often involves illegal activities, such as unauthorized access to systems and data breaches. Even if the cause is perceived as noble, the legality of the actions can be problematic.

4. Transparency and Accountability: Hacktivists often operate anonymously, raising questions about accountability. Without clear accountability, it is difficult to assess the true impact and motives of their actions.

Justification by Gray Hat Hackers

Gray hat hackers occupy a middle ground between white hat (ethical hackers) and black hat (malicious hackers). They may engage in activities that are legally ambiguous but not necessarily malicious, often driven by a belief in a higher ethical or social cause.

A gray hat hacker justifying their actions in the name of activism might argue the following points:

1. Public Interest: They might believe their actions serve the public interest by exposing vulnerabilities, corruption, or unethical practices. By bringing these issues to light, they hope to instigate positive change.

2. Greater Good: Gray hat hackers often operate under the principle of the greater good, asserting that their actions, although illegal or unethical, are justified if they lead to beneficial outcomes for society.

3. Lack of Other Avenues: They might argue that traditional avenues for change, such as petitions or lobbying, are ineffective or unavailable, leaving hacking as a necessary means to achieve their goals.

4. Ethical Responsibility: Some gray hat hackers view it as their ethical responsibility to expose wrongdoings, particularly when powerful entities are involved. They may see themselves as whistleblowers using unconventional methods.

However, the justification of hacktivist actions by gray hat hackers remains a contentious issue. While some view their actions as a form of digital civil disobedience, others see them as reckless and potentially harmful. The ethical landscape of hacktivism is complex and often depends on the perspectives and values of the individuals involved and the society in which they operate.

THANK YOU