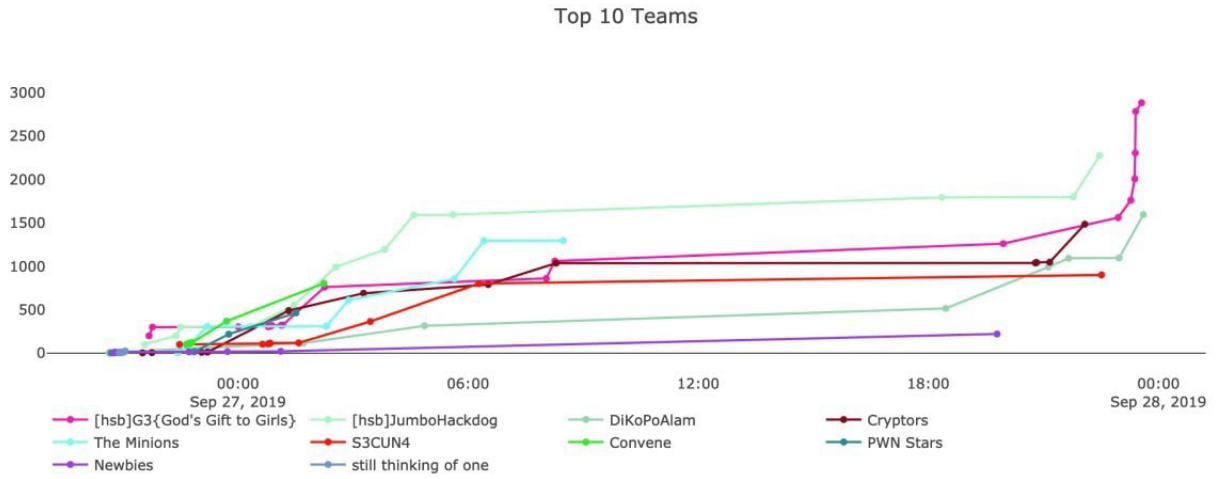
ROOTCON 2019's CTF Writeups for Web Category

Aj Dumanhug Follow
Sep 29 · 6 min read

ROOTCON 13 Official Banner

Introduction

It's my second time to join ROOTCON's Capture the Flag (CTF) competition. Last year, we were lucky to win the 2nd place against more than ten teams. This year, we gave our best to win the CTF, and we've done it! (Yey!)



Place	Team	Score
1	[hsb]G3{God's Gift to Girls}	2884
2	[hsb]JumboHackdog	2276
3	DiKoPoAlam	1596

The Final Scoreboard

On this writeup, we'll show you how we almost wipe the web category for this year's CTF. It's hard and challenging — what a great set of challenges, Pwn De Manila (Thank you!).

Web Challenges

. . .

Winner Winner Chicken Dinner! (100 points)

Landing Page

Solution

- The hyperlink is pointing to https://www.rootcon.org/.
- We need to add a Referer in the request and the value should be the last year winner of Rootcon's CTF competition. Which can be found here: https://www.rootcon.org/html/halloffame

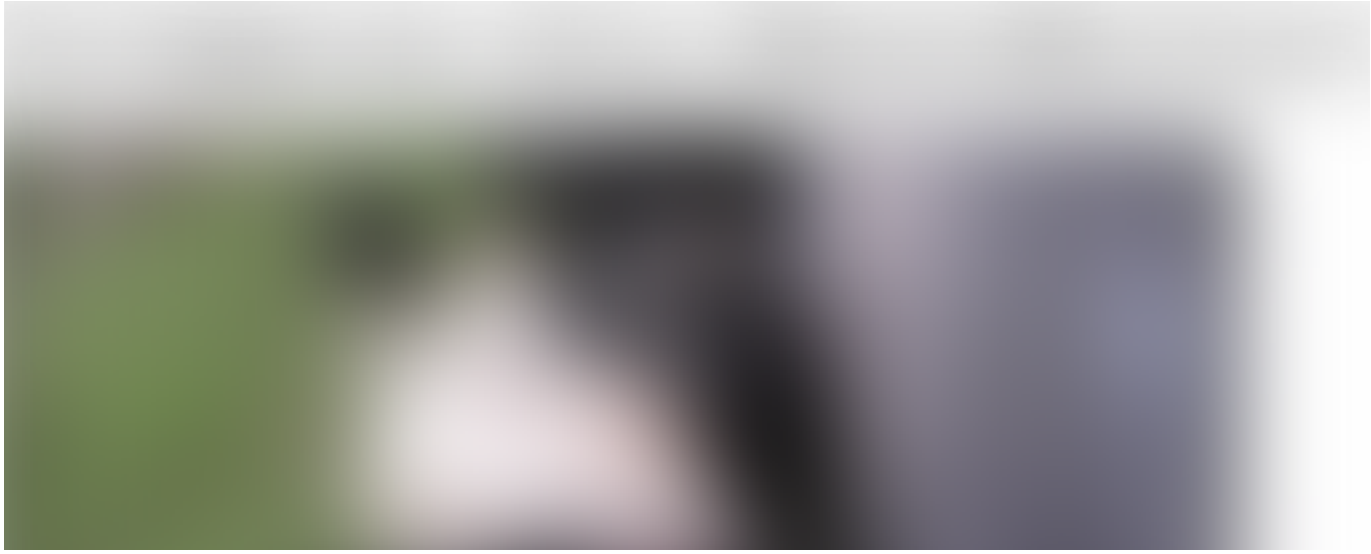
```
curl http://10.0.2.12/ -H "Referer: Harambae"
```

> *The flag is RC13{A_w1nN3r_i5_Y0u!}*

. . .

Khal Dereta (200 points)

Landing Page



Solution

- Enumerate the directories and files of the challenge website. I used dirsearch tool and one of the interesting file is the admin.html

```
[11:23:52] 200 -  187KB - /admin.html
```

- Viewing the source code of the page will greet us with an interesting code which is a JSFuck.

- Converting it to plain text will give us this following code:

```
if (document.forms[0].username.value == "H4ramba3" &&
document.forms[0].password.value == "iwantburritosnotjustic3")
document.location =
"YVq85a5TqK0SprPBUoPBueFPm9cI7IWnt6jxtQffGJjZ7PP7.php"
```

- Login to admin page using **H4ramba3** as the username and **iwantburritosnotjustic3** as the password to get the flag.

> *The flag is RC13{unFvcK_th3_5tr1Ngzzzzz}*

. . .

Cleanliness Notice: Pwn De Manila Employees! (300 points)

Landing Page

Source Code

Solution

- To solve this challenge, we have to send the string `pretty_good_passphrase` but that is not easy because we have to bypass the **preg_replace** function that replaces the phrase with blank value.
- The final payload is `prettypretty_good_passphrase_good_passphrase` because it will remove the `pretty_good_passphase` and leave `pretty` and `_good_passphrase`.

> ***The flag is RC13{iNput_s4n1ta410N_i5_W3aK}***

. . .

M4ny (300 points)

Landing Page

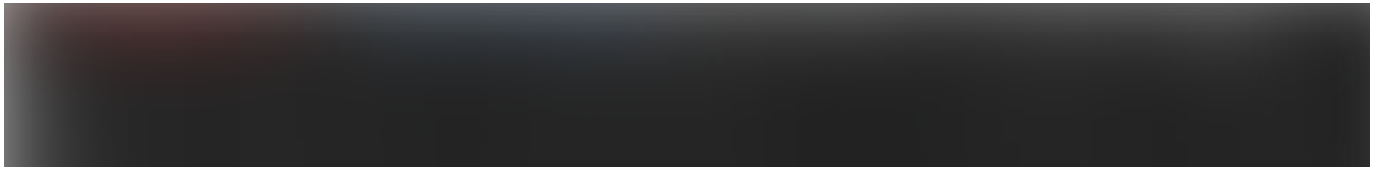
Solution

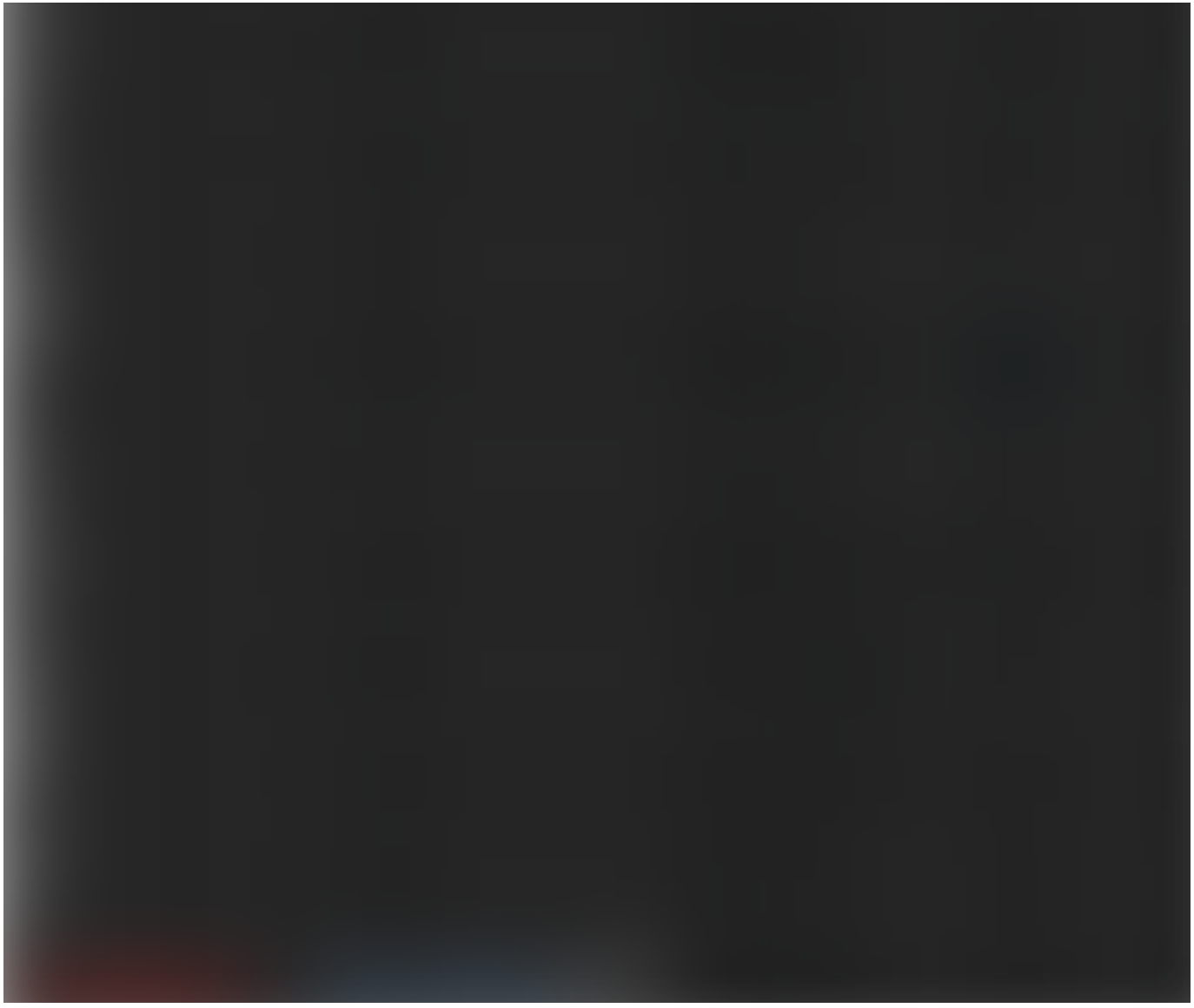
- On this challenge, the hints are the title of the page which is "**Symb0l0gy**", and the "**118+**" & the "**Polonium-Bending**" on the image. These are all related to Periodic Elements. **Polonium** is an element with a symbol of **Po** and the total number of elements in the periodic table is **118**.

- To exploit this, we had to create a wordlist of symbols of all the elements. We used this gist from GitHub: https://gist.github.com/GoodmanSciences/c2dd862cd38f21b0ad36b8f96b4bf1ee
- Then we wrote a python script to automate the requests to get the flag.

```
import os
import requests as req
from bs4 import BeautifulSoup as bs


with open('elements.lst') as f:
    elements = f.readlines()
    for symbol in elements:
        link = "http://10.0.2.14/" + symbol + ".php"
        link = link.replace("\n", "")
        response = req.get(link)
        if response.status_code == 200:
            soup = bs(response.content, 'lxml')
            print(soup.select_one('title').text)
        else:
            continue
```





> *The flag is RC13{th3_eL3mEnt5_w1LL_D3stR0y_y0u!}*

. . .

Friday Madness (500 points)

Landing Page

Source Code

A hint was found in the source code. Visiting the log.txt file will give us the following content.

Solution

- The objective of this challenge is to identify the ID for the following commit:

```
Sun Jun 02 2019 14:21:45        GMT+0800 (UTC)    Added flag
```

- The list of IDs on the landing page has a pattern which gave us an idea on how to craft the ID for the commit above. So during the competition, I read a lot of stuff and found out that the ID is an ObjectID of MongoDB.

The last valid ID is `5ceb5d394a6bd51a08f6cde7`.

Following the documentation of ObjectID, we can easily identify the strings in the ID.

`5ceb5d39` is the 4-byte value representing the seconds since the Unix epoch;

`4a6bd5` is the 3-byte machine identifier;

`1a08` is the 2-byte process id;

`f6cde7` is the 3-byte counter, starting with a random value.

After checking again the list of valid IDs — the **machine identifier** and **process id** should be the same. And we should focus on the first 4-byte and last 3-byte of the ID.

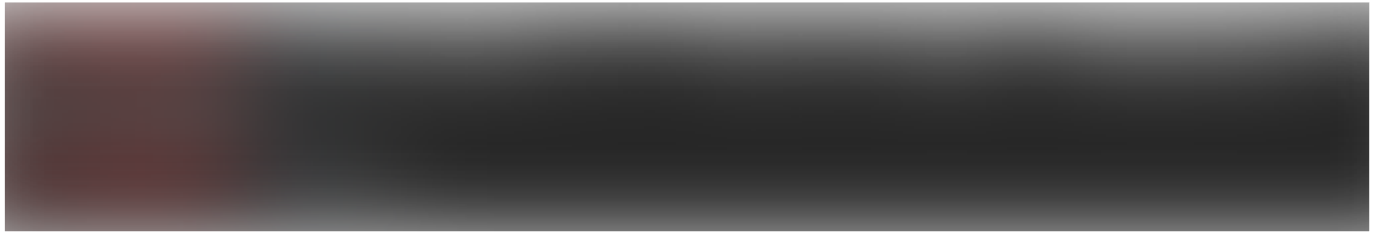
The first 4-byte is the hexadecimal of the seconds of the commit in log.txt.

`5ceb5d394a6bd51a08f6cde7` is for the commit `Mon May 27 2019 11:44:57 GMT+0800 (UTC) Added security implements`.

To validate and convert `5ceb5d39` into Unix timestamp, we can simply perform the following formula:

```
Math.round(new Date("Mon May 27 2019 11:44:57 GMT+0800 (UTC)")/1000)
```

And convert that timestamp to hex:



From here, we can confirm that `5ceb5d39` is a Unix timestamp of the commit.

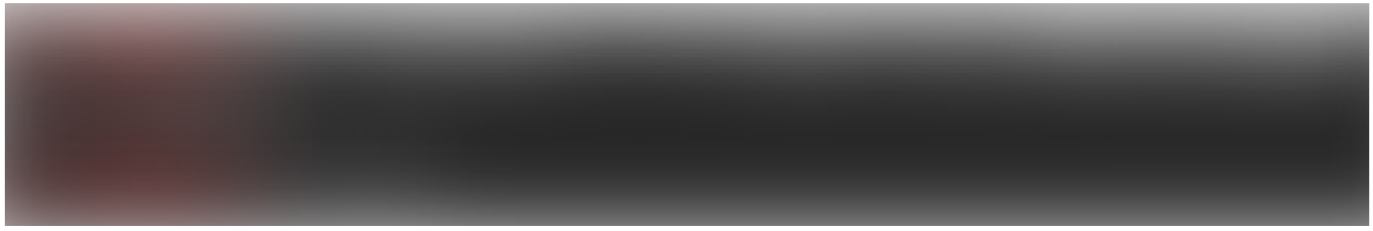
To get the flag for the commit below, we have to craft the ID:

```
Sun Jun 02 2019 14:21:45 GMT+0800 (UTC) Added flag
```

Getting the timestamp:

```
Math.round(new Date("Sun Jun 02 2019 14:21:45 GMT+0800 (UTC)")/1000)
```


Converting it to hex:



Crafting the ObjectID:

First 4-byte is `5cf36af9`
Machine identifier is `4a6bd5`

Process id is `1a08`

`5cf36af94a6bd51a08` + the next value of `f6cde7` which is `f6cde8`

Final ObjectID is `5cf36af94a6bd51a08f6cde8`.

> ***The flag is RC13{tH3se_Ar3_N0t_Th3_0bj3cTs_uR_l0ok1nG_F0r}***

. . .

To end this writeup here is the picture of our team after winning the CTF.

As always, thank you for reading!

. . .

Follow Infosec Write-ups for more such awesome write-ups.

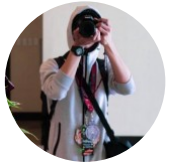
InfoSec Write-ups

A collection of write-ups from the best hackers in the world on topics ranging from bug bounties and CTFs to vulnhub...

medium.com

Web Ctf Rootcon Hackstreetboys Cybersecurity

164 claps



WRITTEN BY

Aj Dumanhug

CTO /CISO at Secuna, Moderator at hackstreetboys, Cybersecurity Trainer at UP and Adamson. Cybersecurity PH CERT and ROOTCON 13 CTF Champion.

Follow

InfoSec Write-ups

Follow

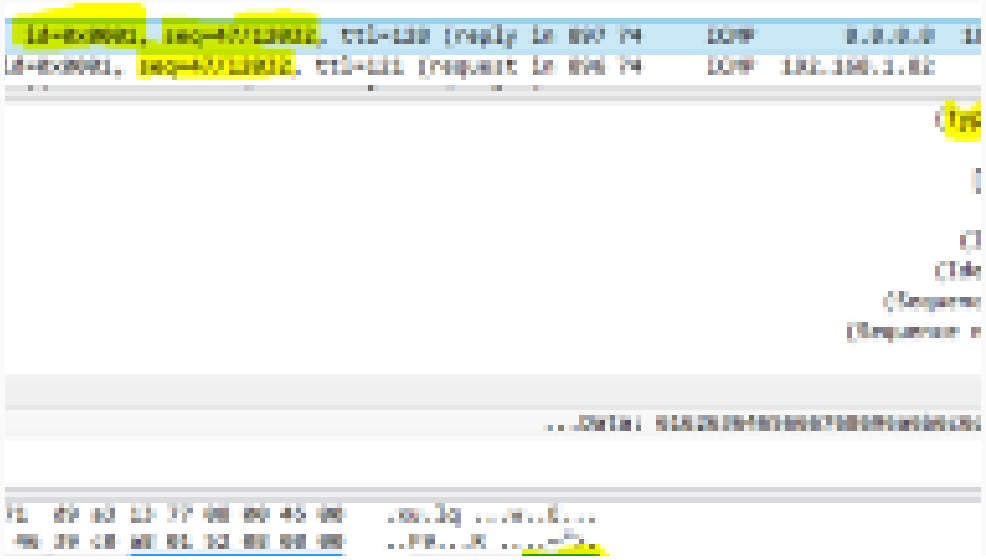
A collection of write-ups from the best hackers in the world on topics ranging from bug bounties and CTFs to vulnhub machines, hardware challenges and real life encounters. In a nutshell, we are the largest InfoSec publication on Medium. Powered by Hackrew

Write the first response

More From Medium

More from InfoSec Write-ups

Ping Power — ICMP Tunnel



Nir Chako in InfoSec Write-ups
Dec 17, 2018 · 8 min read

1.1K

More from InfoSec Write-ups

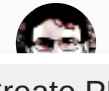
Picture Yourself Becoming a Hacker Soon (Beginner's Guide)

Abanikanda in InfoSec Write-ups
Aug 16 · 16 min read ★

483

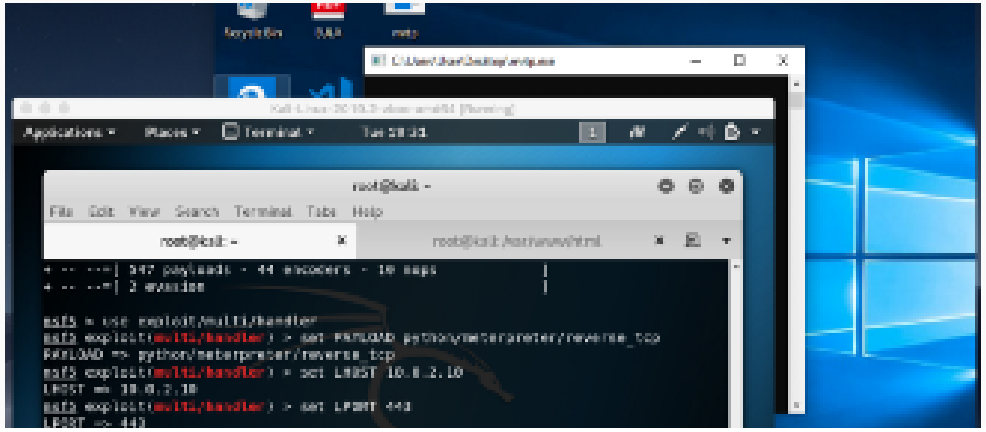
More from InfoSec Write-ups

Antivirus Evasion with Python



Marcelo Sacchetin in InfoSec Write-ups

610



Jun 11 · 6 min read ★