

PERSIAPAN LKS

REJA REVALDY F.

COMPLETED

- Crackme1
- Petiharta
- Reverse1
- Reverse2
- Whoami

## Crackme1

1. Diberikan sebuah program dan kita harus menginputkan sebuah password, lalu saya langsung saja menggunakan ltrace dan menemukan strcmp inputan kita dibandingkan dengan sebuah string "JJJJJJJJJJJJJBxs", lalu saya langsung saja coba inputkan string tersebut dan menemukan flagnya

```

(kali@kali)-[~/./exercise/Persiapan/reverse/crackme1]
└─$ ltrace ./crackme
puts("Hi!\nInput Your Password"Hi!
Input Your Password
)
malloc(18)
memset(0x55ea099616b0, '\0', 18)
fgets(JJJJJJJJJJJJJJBxs
"JJJJJJJJJJJJJJJBxs", 18, 0x7fd8c15d49a0)
strcpy("JJJJJJJJJJJJJJJBxs", "JJJJJJJJJJJJJJJBxs")
printf("MANTUL, flag is LKSSMK28[%s]\n", "JJJJJJJJJJJJJJJBxs"
)
free(0x55ea099616b0)
+++ exited (status 0) +++
= 24
= 0x55ea099616b0
= 0x55ea099616b0
= 0x55ea099616b0
= 0
= 44
= <void>

(kali@kali)-[~/./exercise/Persiapan/reverse/crackme1]
└─$ ./crackme
Hi!
Input Your Password
JJJJJJJJJJJJJJJBxs
MANTUL, flag is LKSSMK28{JJJJJJJJJJJJJJJBxs}

```

**Flag = LKSSMK28{JJJJJJJJJJJJJJJBxs}**

## Petiharta

1. Diberikan sebuah program yang berisikan pencarian harta, lalu saya coba menjalankan program strings untuk melihat library c yang dipakai disini saya menemukan gets yang dimana gets ini rentan di buffer overflow, lalu saya coba melakukan buffer overflow menggunakan python

[illegible]

2. Disini ditemukan bahwa program akan terjadi buffer overflow setelah inputan ke 72 lalu saya coba inputkan 73 string dan string untuk flag nya muncul

FLAG = palui{harta\_yg\_paling\_berharga\_adalah\_ILMU}

## Reverse1

1. Disini kita diberikan sebuah program yang berisikan inputan untuk password lalu saya menggunakan cara yang sama dengan crackme1 dan saya mendapatkan flagnya

```
(kali@kali)~/../exercise/Persiapan/reverse/reverse1
$ ltrace ./reverse1
_ZNSt8ios_base4InitC1Ev(0x558ca88f62b9, 0xffff, 0x7ffdc354df38, 224) = 0x7f0a8288980
__cxa_atexit(0x7f0a829445a0, 0x558ca88f62b9, 0x558ca88f6060, 6) = 0
strcpy(0x7ffdc354dce3, "k0o") = 0x7ffdc354dce3
strcat("k0o", "pi_h") = "k0opi_h"
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 4, 0x685f69)
) = 0x558ca88f6080
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f4010, 0, 3072) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 3072) = 0x558ca88f6080
) = 0x558ca88f6080
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f4039, 0, 3072) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 3072) = 0x558ca88f6080
) = 0x558ca88f6080
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f4058, 0, 3072) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 3072) = 0x558ca88f6080
) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 3072) = 0x558ca88f6080
) = 0x558ca88f6080
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f4077, 0, 3072) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 3072) = 0x558ca88f6080
) = 0x558ca88f6080
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f4085, 0, 3072) = 0x558ca88f6080
_ZStsrISt11char_traitsIcEERSt13basic_istreamIcT_ES6_PS3_(0x558ca88f61a0, 0x7ffdc354dd20, 0x7f0a82a8e3b0, 0x203e3a64726f7773) = 0x558ca88f61a0
password: dsadasd
) = 0x558ca88f61a0
strcat("k0opi_h", "ita") = "k0opi_hita"
strcat("k0opi_hita", "m_pht") = "k0opi_hitam_pht"
strcmp("dsadasd", "k0opi_hitam_pht") = -7
_ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc(0x558ca88f6080, 0x558ca88f409f, 107, 0xffffffff) = 0x558ca88f6080
_ZNSoLsEPFRSo5_E(0x558ca88f6080, 0x7f0a829b8010, 0x558ca88f6080, 0x202141534942204b) = 0x558ca88f6080
) = 0x558ca88f6080
+++ exited (status 0) +++

(kali@kali)~/../exercise/Persiapan/reverse/reverse1
$ ./reverse1

.CTF: ` .LKS-SMK28`

CTF LKS SMK28
password:> k0opi_hitam_pht
LKSSMK28{01c9fsd3gt34zxxcb0eb8a42d3c534rf3c570703e3t}
```

FLAG = LKSSMK28{01c9fsd3gt34zxxcb0eb8a42d3c534rf3c570703e3t}

## Reverse2

1. Disini kita diberikan sebuah program yang berisikan inputan untuk password lalu saya menggunakan cara yang sama dengan reverse1 dan saya mendapatkan flagnya

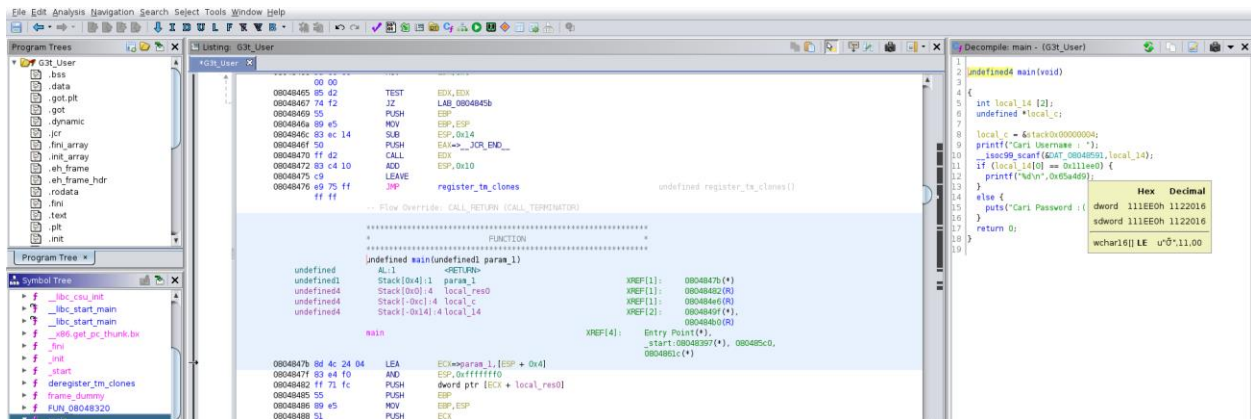
```
(kali@kali)-[~/exercise/Persiapan/reverse/reverse2]
$ ltrace ./reverse2
puts("||=====|...|=====|")
)
= 73
puts("||=====|...|=====|")
)
= 73
puts("||=====| CT...|()=====| CTF |=====|()|")
)
= 66
puts("||=====| LKS SM...|()=====| LKS SMK 28=====|()|")
)
= 66
puts("||=====|...|=====|")
)
= 73
puts("||=====|...|=====|")
)
= 73
puts("Password:Password:
)
= 10
__isoc99_scanf(0x55df9df6713c, 0x7fff921ef6a0, 0, 0x7f26dcefa963asdas
)
= 1
strcmp("0x00007fff", "asdas")
= -49
puts("You FailedYou Failed
)
= 11
+++ exited (status 0) +++

(kali@kali)-[~/exercise/Persiapan/reverse/reverse2]
$ ./reverse2
||=====| | |
||=====|
||=====| CTF |=====|
||=====| LKS SMK 28=====|
||=====|
Password:
0x00007fff
You Win
LKSSMK28{LKSSMK28_486619254b9c9f6e6800cfae77}
```

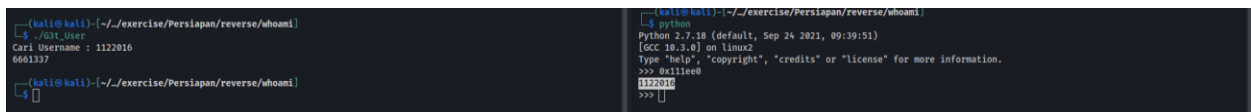
FLAG = LKSSMK28{LKSSMK28\_486619254b9c9f6e6800cfae77}

## Whoami

1. Diberikan sebuah program yang memiliki fungsi mencari username lalu memunculkan password, setelah saya menggunakan ltrace saya tidak menemukan apa apa lalu saya coba ghidra untuk melihat pseudo code program ini lalu saya menemukan kondisi dimana jika inputan kita sama dengan hex yang diberikan maka akan memunculkan password nya



2. Disini saya langsung coba masukan hex tersebut dan gagal lalu saya coba ubah ke decimal dan berhasil memunculkan password dari user tersebut



FLAG = {1122016}