KKSI

KOMPETISI KOMUNITAS SIBER INDONESIA

[ONLINE - PENYISIHAN]

TNI - JAKARTA

Nama Team : IoT

Ketua Team

Didit Dwi Aftianto

Anggota

Rusly Tomagola
Abdullah Iksan
Zaenudin

3 November 2019

Android-TNI

Download :

https://drive.google.com/file/d/1YavIgKoaG0GF0ZyUYYaxc0BfUD9K6814/view

Diberikan sebuah file APK bernama _KKSI_.apk, selanjutnya melakukan verifikasi file **_KKSI_.apk**

$ **file _KKSI_.apk**

KKSI.apk: Zip archive data, at least v2.0 to extract

APK memang hanya sebuah arsip berbentuk ZIP dan format dasarnya adalah JAR. Tetapi isi sebenarnya dari arsip .zip disusun sedemikian rupa untuk menjadi aplikasi yang sebenernya karna didalamnya tersimpan berbagai macam file.

Banyak cara untuk membuka file tersebut.

Langkah 1, membuka dengan menggunakan **apktool** (a tool for reengineering Android apk files)

```
Apktool v2.4.0 - a tool for reengineering Android apk files
with smali v2.2.6 and baksmali v2.2.6
Copyright 2014 Ryszard Wiśniewski <brut.alll@gmail.com>
Updated by Connor Tumbleson <connor.tumbleson@gmail.com>

usage: apktool
 -advance,--advanced   prints advance information.
 -version,--version    prints the version then exits
usage: apktool if|install-framework [options] <framework.apk>
 -p,--frame-path <dir>   Stores framework files into <dir>.
 -t,--tag <tag>          Tag frameworks using <tag>.
usage: apktool d[ecode] [options] <file_apk>
 -f,--force              Force delete destination directory.
 -o,--output <dir>       The name of folder that gets written. Default is apk.out
 -p,--frame-path <dir>   Uses framework files located in <dir>.
 -r,--no-res             Do not decode resources.
 -s,--no-src             Do not decode sources.
 -t,--frame-tag <tag>    Uses framework files tagged by <tag>.
usage: apktool b[uild] [options] <app_path>
 -f,--force-all          Skip changes detection and build all files.
 -o,--output <dir>       The name of apk that gets written. Default is dist/name.apk
 -p,--frame-path <dir>   Uses framework files located in <dir>.

For additional info, see: http://ibotpeaches.github.io/Apktool/
For smali/baksmali info, see: https://github.com/JesusFreke/smali
```

$ **apktool d _KKSI_.apk -o androtni/**

```
I: Using Apktool 2.4.0 on _KKSI_.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /Users/dontukulesto/Library/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Baksmaling classes2.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

setelah selesai mendecode file **_KKSI _.apk** tersebut, maka dapat dilihat pada folder **androtni** yang berisikan **1188 direktori dan 17399 berkas**.

Tidak ada informasi tambahan pada soal, hanya "**Like, share and get the flag**" dan itu akan kita jadikan sebagai kata kunci pencarian didalam folder dan file dengan menggunakan perintah **grep** pada direktori kerja **androtni** dengan beberapa tambahan perintah.

-n untuk menampilkan nomer line pada kata yang dicari.

-r untuk mencari didalam folder.

. adalah direktori kerja.

Dimulai dari kata kunci pertama "**Like**"

$ **grep -nr "like*" . | more**

tidak ada hasil yang sesuai keinginan

$ **grep -nr "share*" . | more**

```
./res/values-zh/strings.xml:37:    <string name="shareSiteSubject">网站分享</string>
./res/values-uk/strings.xml:17:    <string name="abc_shareactionprovider_share_with">Надіслати через</string>
./res/values-uk/strings.xml:18:    <string name="abc_shareactionprovider_share_with_application">Поділитися через додаток %s</string>
./res/values-uk/strings.xml:64:    <string name="shareContentSubject">Get this cool Android App</string>
./res/values-sr/strings.xml:17:    <string name="abc_shareactionprovider_share_with">Дели са</string>
./res/values-sr/strings.xml:18:    <string name="abc_shareactionprovider_share_with_application">Дељење са апликацијом %s</string>
./res/values-sr/strings.xml:64:    <string name="shareContentSubject">Get this cool Android App</string>
./res/raw/configuration.xml:38: <shareExtraLink>KKSI2019{7093571a3d4d9f735126b252a332eb6a}</shareExtraLink>
./res/values-pa/strings.xml:17:    <string name="abc_shareactionprovider_share_with">ਇਸ ਰਾਹ ਸਾਂਝਾ ਕਰੋ</string>
./res/values-pa/strings.xml:18:    <string name="abc_shareactionprovider_share_with_application">%s ਰਾਹ ਸਾਂਝਾ ਕਰੋ</string>
./res/values-si/strings.xml:17:    <string name="abc_shareactionprovider_share_with">බෙදා ගන්න</string>
./res/values-si/strings.xml:18:    <string name="abc_shareactionprovider_share_with_application">%s බෙදා ගන්න</string>
./res/values-b+sr+Latn/strings.xml:17:    <string name="abc_shareactionprovider_share_with">Deli sa</string>
./res/values-b+sr+Latn/strings.xml:18:    <string name="abc_shareactionprovider_share_with_application">Deljenje sa aplikacijom %s</string
>
```

kata kunci "**share**" tersebut membuahkan hasil meskipun harus sangat teliti membacanya.

./res/raw/configuration.xml:38:

<shareExtraLink>KKSI2019{7093571a3d4d9f735126b252a332eb6a}</s

hareExtraLink>

kalimat tersebut terdapat pada subdirektori **res** yaitu **raw** dengan namafile **configuration.xml** dan terletak pada line **38**

Langkah 2, membuka file dengan **7-Zip.**

x adalah perintah untuk mengeluarkan file

-o adalah nama direktori untuk hasil keluar file

$ **7za x _KKSI_.apk -oandroidtni/**

```
7-Zip (a) [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=utf8,Utf16=on,HugeFiles=on,64 bits,4 CPUs x64)

Scanning the drive for archives:
1 file, 14265377 bytes (14 MiB)

Extracting archive: _KKSI_.apk
--
Path = _KKSI_.apk
Type = zip
Physical Size = 14265377

Everything is Ok

Files: 1110
Size:       28705148
Compressed: 14265377
```

$ **grep -nr "share*" . | more**

```
Binary file ./resources.arsc matches
Binary file ./res/anim/abc_popup_enter.xml matches
Binary file ./res/anim/abc_grow_fade_in_from_bottom.xml matches
Binary file ./res/anim/abc_shrink_fade_out_from_bottom.xml matches
Binary file ./res/anim/abc_popup_exit.xml matches
./res/raw/configuration.xml:38: <shareExtraLink>KKSI2019{7093571a3d4d9f735126b252a332eb6a}</shareExtraLink>
./META-INF/MANIFEST.MF:249:Name: res/drawable-hdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/MANIFEST.MF:441:Name: res/drawable-hdpi/outline_share_black_24.png
./META-INF/MANIFEST.MF:519:Name: res/drawable-mdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/MANIFEST.MF:603:Name: res/drawable-mdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/MANIFEST.MF:630:Name: res/drawable-xhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/MANIFEST.MF:899:Name: res/drawable-hdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/MANIFEST.MF:950:Name: res/drawable-mdpi/outline_share_black_24.png
./META-INF/MANIFEST.MF:1152:Name: res/drawable-xxhdpi/outline_share_black_24.png
./META-INF/MANIFEST.MF:1528:Name: res/drawable-xhdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/MANIFEST.MF:1693:Name: res/drawable-hdpi/ic_share_black_24dp.png
./META-INF/MANIFEST.MF:1836:Name: res/drawable-mdpi/ic_share_black_24dp.png
./META-INF/MANIFEST.MF:1945:Name: res/drawable-xxxhdpi/outline_share_black_24.png
./META-INF/MANIFEST.MF:2315:Name: res/drawable-xhdpi/ic_share_black_24dp.png
./META-INF/MANIFEST.MF:2387:Name: res/drawable-xxhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/MANIFEST.MF:2483:Name: res/drawable-xxxhdpi/ic_share_black_24dp.png
./META-INF/MANIFEST.MF:2729:Name: res/drawable-xhdpi/outline_share_black_24.png
./META-INF/MANIFEST.MF:2795:Name: res/drawable-xxxhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/MANIFEST.MF:3075:Name: res/drawable-xxhdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/MANIFEST.MF:3338:Name: res/drawable-xxhdpi/ic_share_black_24dp.png
./META-INF/ANDROID_.SF:251:Name: res/drawable-hdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/ANDROID_.SF:443:Name: res/drawable-hdpi/outline_share_black_24.png
./META-INF/ANDROID_.SF:521:Name: res/drawable-mdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/ANDROID_.SF:605:Name: res/drawable-mdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/ANDROID_.SF:632:Name: res/drawable-xhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/ANDROID_.SF:901:Name: res/drawable-hdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/ANDROID_.SF:952:Name: res/drawable-mdpi/outline_share_black_24.png
./META-INF/ANDROID_.SF:1154:Name: res/drawable-xxhdpi/outline_share_black_24.png
./META-INF/ANDROID_.SF:1530:Name: res/drawable-xhdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/ANDROID_.SF:1695:Name: res/drawable-hdpi/ic_share_black_24dp.png
./META-INF/ANDROID_.SF:1838:Name: res/drawable-mdpi/ic_share_black_24dp.png
./META-INF/ANDROID_.SF:1947:Name: res/drawable-xxxhdpi/outline_share_black_24.png
./META-INF/ANDROID_.SF:2317:Name: res/drawable-xhdpi/ic_share_black_24dp.png
./META-INF/ANDROID_.SF:2389:Name: res/drawable-xxhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/ANDROID_.SF:2485:Name: res/drawable-xxxhdpi/ic_share_black_24dp.png
./META-INF/ANDROID_.SF:2731:Name: res/drawable-xhdpi/outline_share_black_24.png
./META-INF/ANDROID_.SF:2797:Name: res/drawable-xxxhdpi/abc_ic_menu_share_mtrl_alpha.png
./META-INF/ANDROID_.SF:3077:Name: res/drawable-xxhdpi/abc_ab_share_pack_mtrl_alpha.9.png
./META-INF/ANDROID_.SF:3340:Name: res/drawable-xxhdpi/ic_share_black_24dp.png
Binary file ./classes2.dex matches
Binary file ./classes.dex matches
Binary file ./lib/armeabi-v7a/libmodpdfium.so matches
Binary file ./lib/arm64-v8a/libmodpdfium.so matches
```

FLAG : **KKSI2019{7093571a3d4d9f735126b252a332eb6a}**

Crypto

OBFUS-TNI

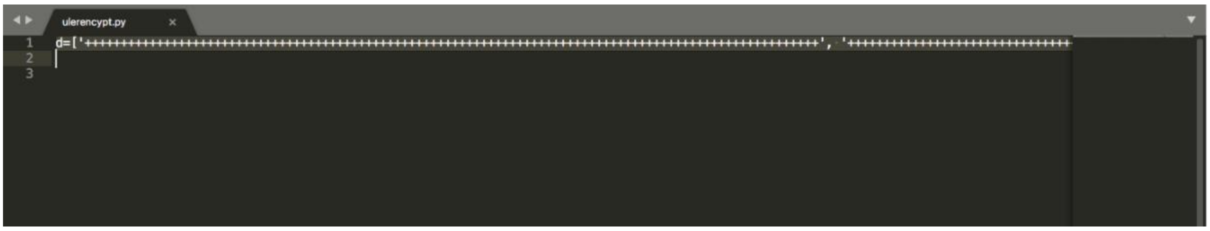
Can Pyton code be Protected

Download file di

https://drive.google.com/open?id=1S3XTZ8ZKJlG_TfFOCyctLE5ClK4EFS9G

Format Flag : KKSI2019{flag}

Setelah di download dan di buka dengan arsip manager terdapat 1 buah script python yang terproteksi dalam 1 line agar sulit dibaca.

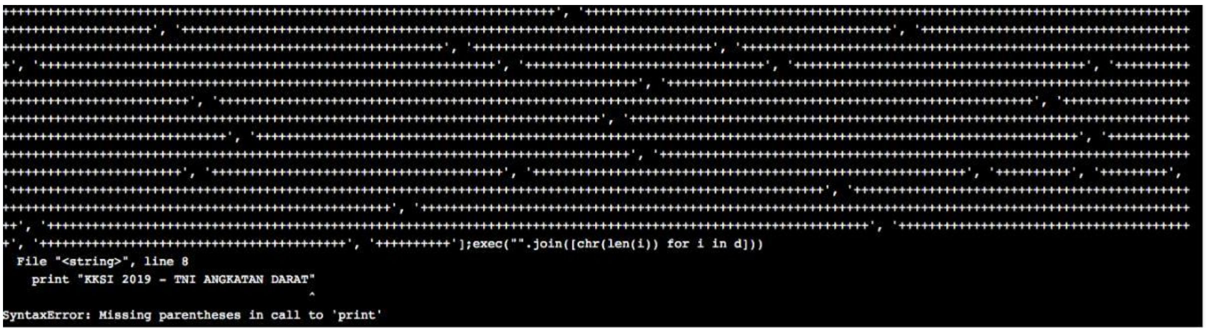


Ketika di jalankan akan muncul sebuah pesan error

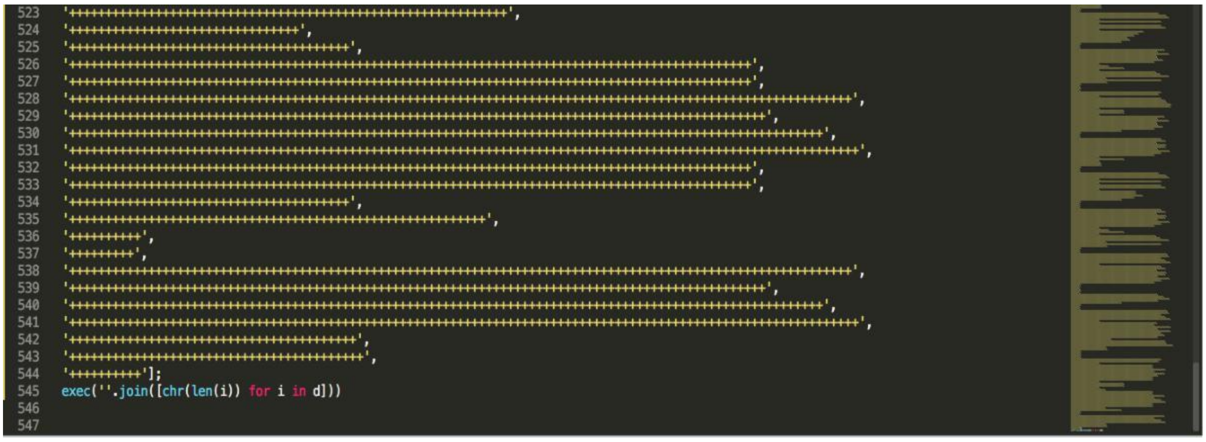
File "main.py", line 8

print "KKSI 2019 - TNI ANGKATAN DARAT"

SyntaxError: Missing parentheses in call to 'print'



Mencoba untuk merapihkhan kode tersebut secara manual juga tetap akan muncul pesan error tersebut.



Mencoba merapihkan dengan Stream Editor yang ada pada linux untuk melakukan pergantian teks manipulasi tanpa harus membukanya.

Dengan perintah sebagai berikut:

$ cat ulerencypt.py | sed 's/exec/print/' | python

Hasil dari perintah tersebut:

```
from datetime import datetime

KKSI = [75, 75, 83, 73, 50, 48, 49, 57, 123, 101, 55, 98, 102, 52, 100, 100, 53, 49, 97, 101, 53, 57, 99, 97]
TNIAD = [49, 53, 54, 101, 102, 55, 52, 56, 99, 48, 100, 49, 48, 48, 56, 57, 99, 125]


def main():
	print "KKSI 2019 - TNI ANGKATAN DARAT"
	input = raw_input("Cek Waktu: ").strip()
	merdeka = ""
	for i in TNIAD:
		merdeka += chr(i)

	if input == merdeka:
		indonesia = ""
		for i in KKSI:
			indonesia += chr(i)
		print indonesia

	else:
		print datetime.now()


if __name__ == '__main__':
	main()
```

Setelah mendapatkan kode asli dari script python yang terproteksi, jika kembali kepada pesan error pada line 8 yang muncul sebelum kode tersebut dirapihkan, maka telihat jelas ada beberapa fungsi yang salah dalam penulisan.

```python
from datetime import datetime

KKSI = [75, 75, 83, 73, 50, 48, 49, 57, 123, 101, 55, 98, 102, 52, 100, 100, 53, 49, 97, 101, 53, 57, 99, 97]
TNIAD = [49, 53, 54, 101, 102, 55, 52, 56, 99, 48, 100, 49, 48, 48, 56, 57, 99, 125]


def main():
    print "KKSI 2019 - TNI ANGKATAN DARAT"
    input = raw_input("Cek Waktu: ").strip()
    merdeka = ""
    for i in TNIAD:
        merdeka += chr(i)

    if input == merdeka:
        indonesia = ""
        for i in KKSI:
            indonesia += chr(i)
        print indonesia

    else:
        print datetime.now()



if __name__ == '__main__':
    main()

```

setelah script tersebut dirapihkan maka fungsi-fungsi pemanggilan kode python tersebut berjalan sempurna.

```python
from datetime import datetime

KKSI = [75, 75, 83, 73, 50, 48, 49, 57, 123, 101, 55, 98, 102, 52, 100, 100, 53, 49, 97, 101, 53, 57, 99, 97]
TNIAD = [49, 53, 54, 101, 102, 55, 52, 56, 99, 48, 100, 49, 48, 48, 56, 57, 99, 125]


def main():
    print "KKSI 2019 - TNI ANGKATAN DARAT"
    input = raw_input("Cek Waktu: ").strip()

    merdeka = ""
    indonesia = ""

    if input == "merdeka":
        for i in TNIAD:
            merdeka += chr(i)
        print merdeka
    elif input == "indonesia":
        for i in KKSI:
            indonesia += chr(i)
        print indonesia
    else:
        print datetime.now()

if __name__ == '__main__':
    main()

```

jika menginputkan kata **merdeka**, maka fungsi pemanggilan akan mendecode charcode yang berada dalam **TNIAD**, jika menginputkan kata **indonesia**, maka fungsi pemanggilan akan mendecode charcode yang berada dalam **KKSI**.

$ python ulerencypt.py

KKSI 2019 - TNI ANGKATAN DARAT
Cek Waktu: indonesia
KKSI2019{e7bf4dd51ae59ca

$ python ulerencypt.py

KKSI 2019 - TNI ANGKATAN DARAT
Cek Waktu: merdeka
156ef748c0d10089c}

FLAG : **KKSI2019{e7bf4dd51ae59ca156ef748c0d10089c}**

REVERSING

Bongkar Password - TNI

Diberikan sebuah file binary bernama **rev_word** dengan hasil verifikasinya

rev_word: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 3.2.0, BuildID[sha1]=a3357176a9630f7e655d9b11e041d96c4e87f688, not stripped

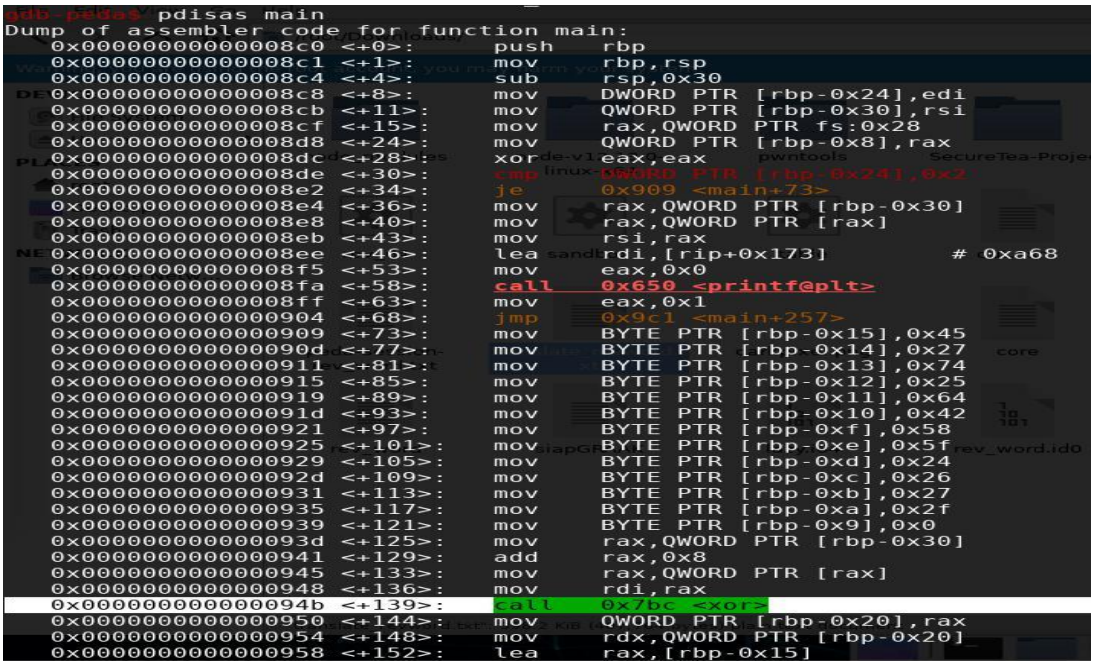
ELF binary-format yang diberikan menggunakan arsitektur 64-bit, agar dapat di jalankan maka harus merubah modenya dengan **chmod a+x rev_word**. Ketika program dijalankan mengeluarkan sebuah pesan:

$ **./rev_word**

Flag: ./rev_word [Cari Dulu Ya :D]
Selanjutnya melakukan analisa statis terhadap binary ELF tersebut dengan **gdb**

$ **gdb -q rev_word**

Selanjutnya melakukan **disasembly** pada fungsi **main**



pada instruksi **lea** (Load Effective Address) merupakan destinasi pemanggilan kode yang harus diinputkan dengan memanggil fungsi **xor** yang menyimpan sebuah kode, dengan kata lain jika ingin menjalankan program **ELF** tersebut harus serta menginputkan kode yang benar terdapat pada instruksi **mov** yang merupakan instruksi untuk menyalin kode ke tujuan yang diinputkan tanpa mempengaruhi sumbernya.

0x0000000000000909 <+73>: mov BYTE PTR [rbp-0x15],**0x45**

0x000000000000090d <+77>: mov BYTE PTR [rbp-0x14],**0x27**

0x0000000000000911 <+81>: mov BYTE PTR [rbp-0x13],**0x74**

0x0000000000000915 <+85>: mov BYTE PTR [rbp-0x12],**0x25**

0x0000000000000919 <+89>: mov BYTE PTR [rbp-0x11],**0x64**

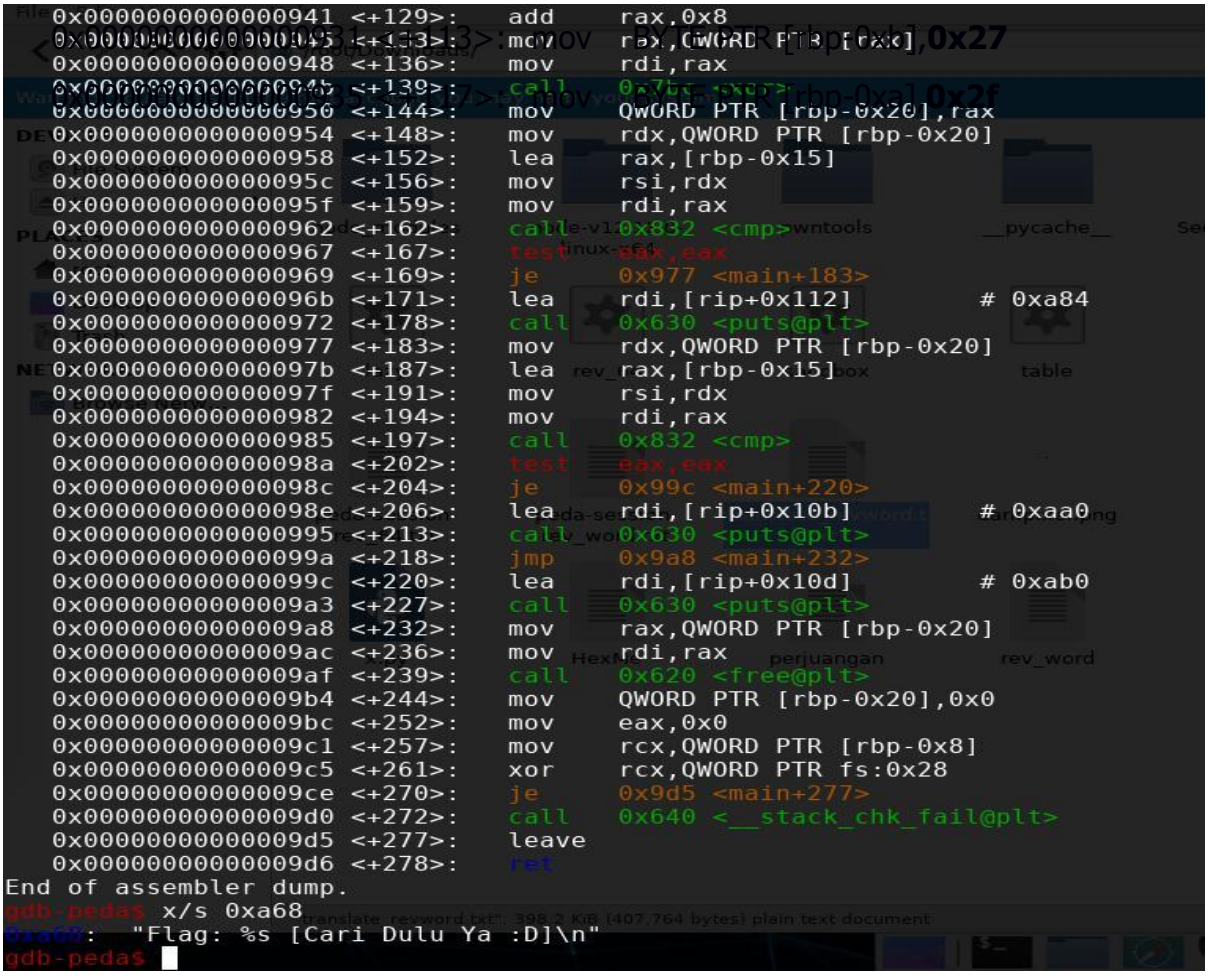
0x000000000000091d <+93>: mov BYTE PTR [rbp-0x10],**0x42**

0x0000000000000921 <+97>: mov BYTE PTR [rbp-0xf],**0x58**

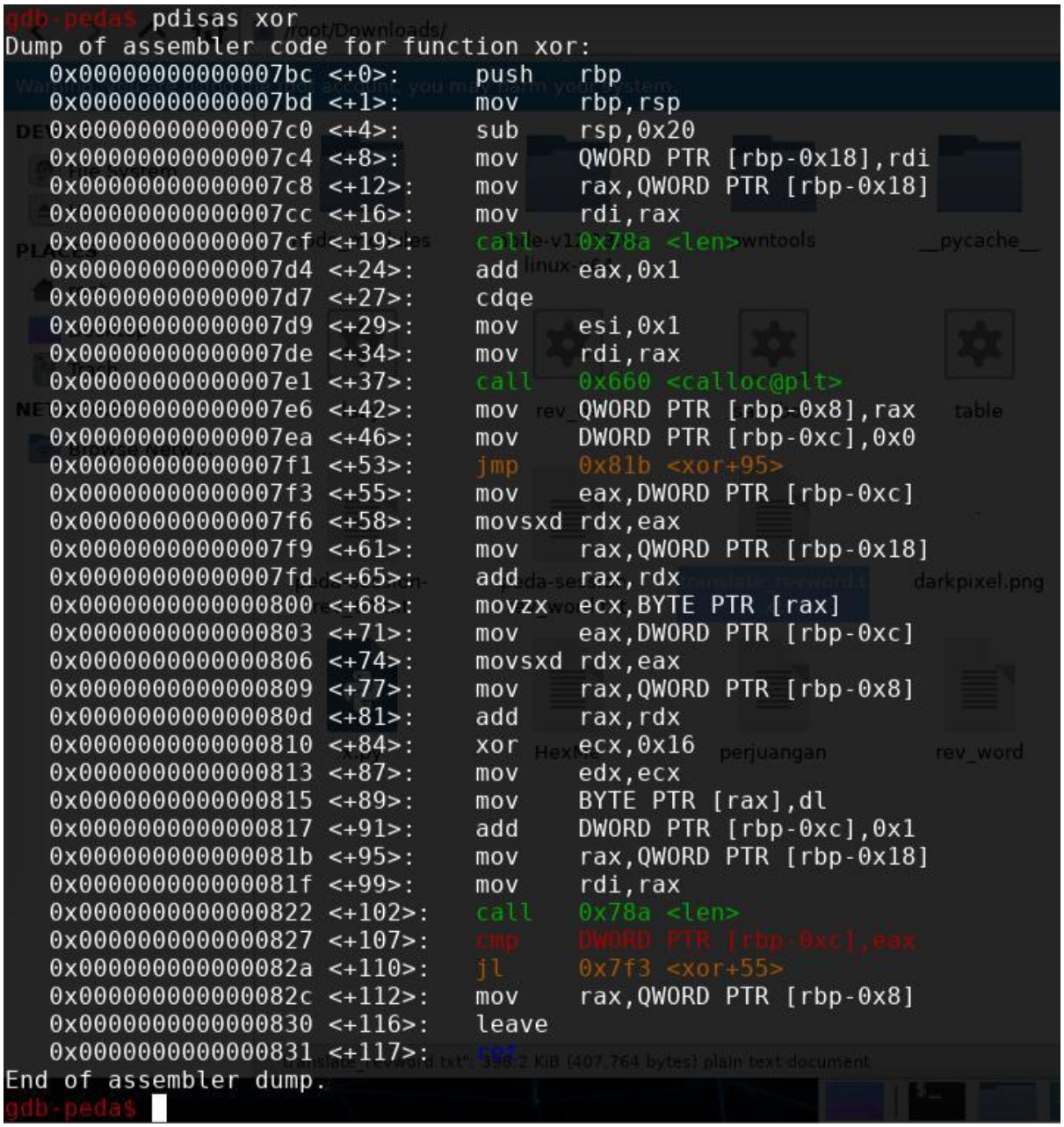
0x0000000000000925 <+101>: mov BYTE PTR [rbp-0xe],**0x5f**

0x0000000000000929 <+105>: mov BYTE PTR [rbp-0xd],**0x24**

0x000000000000092d <+109>: mov BYTE PTR [rbp-0xc],**0x26**



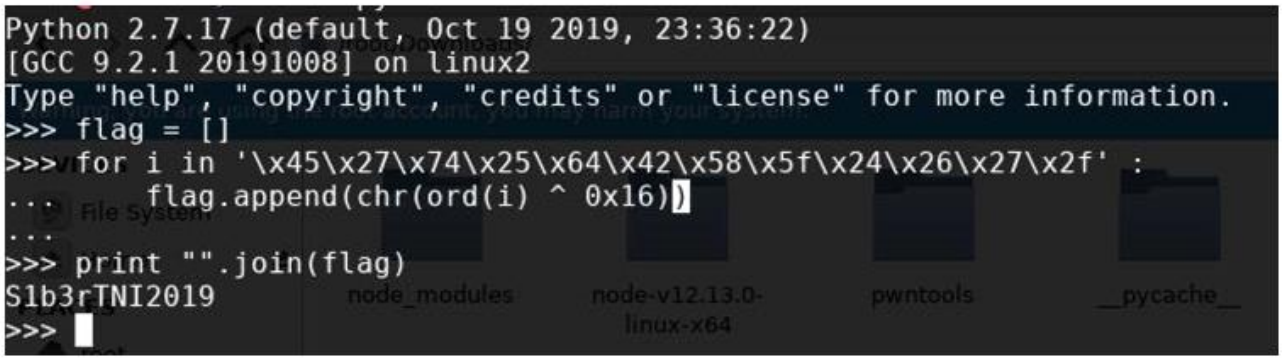
Analisa selanjutnya dilakukan pada fungsi **xor** yang digunakan untuk mengenkripsi kode yang tersimpan pada memory tersebut. Instruksi **XOR** menghubungkan dua nilai menggunakan logika eksklusif **OR**.



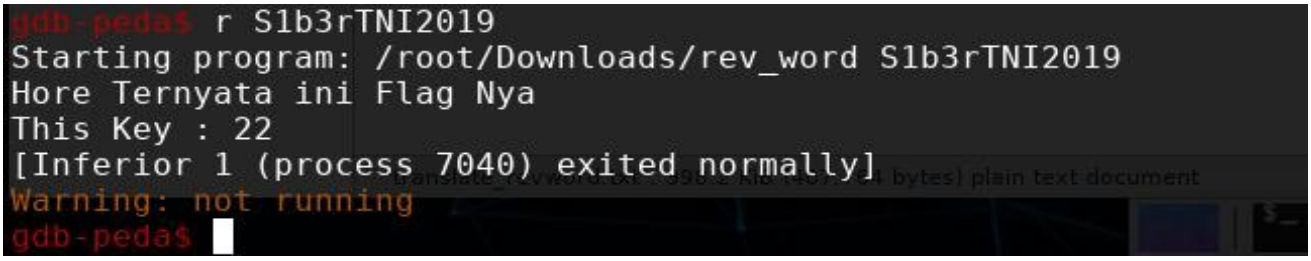
Pada instruksi **xor** terdapat **ecx (EXTENDED COUNT REGISTER)** dari hasil disassemble diketahui akan melakukan looping dan melakukan xor pada setiap looping-nya dengan nilai 0x16

0x0000000000000810 <+84>: xor ecx,0x16

Untuk mendapatkan flag, kita dapat melakukan xor ulang terhadap nilai diatas dengan nilai 0x16 setiap byte-nya.



Menghasilkan sebuah kalimat **S1b3rTNI2019**, setelah itu dimasukkan menjadi sebuah jawaban flag.



FLAG : **KKSI2019{S1b3rTNI2019}**