

# Write Up 0xcbecabe



Ariya Adinatha

Fransiskus Febryan Suryawan

Josep Marcello

# SecNet Anomaly

## Soal

"Tim Incident Response & Forensics dari perusahaan PETIR baru saja mendapatkan kasus terbaru dari salah satu perusahaan digital abal-abal berkedok Cybersecurity bernama SecNet, yang terbukti melakukan scam kepada masyarakat awam dan karyawan bawahannya.

Salah satu karyawan yang BARU SAJA dilantik menjadi karyawan senior SecNet, Noordin, berhasil ditangkap oleh tim PETIR dan diinterogasi. Ia dipercaya menjadi tangan kanan barunya PIC Database Administrator di SecNet. Tim PETIR juga berhasil mengakuisisi sejumlah memori berisikan berkas penting dari laptop Noordin.

Dapatkah kalian membantu Tim PETIR untuk MENDAPATKAN SEJUMLAH PASSWORD apapun yang berkaitan dengan Noordin, baik akun Login Karyawannya, akun Googlenya ataupun akun Githubnya?

Diketahui bahwa Username Noordin bernama Lebah\_g4nteng.

Flag terdiri dari 3 bagian (semuanya akan didapatkan ketika kalian berhasil mengekstrak semua password yang berkaitan dengan Noordin).

Format Flag: CSCCTF{....}

Author: Aseng

## Flag

CSCCTF{4nd\_th1s\_w3\_c0nclude\_that\_the\_w3akest\_link\_of\_53cNet\_15\_h00man\_so\_u\_c4nt\_trust\_em\_4ll! }

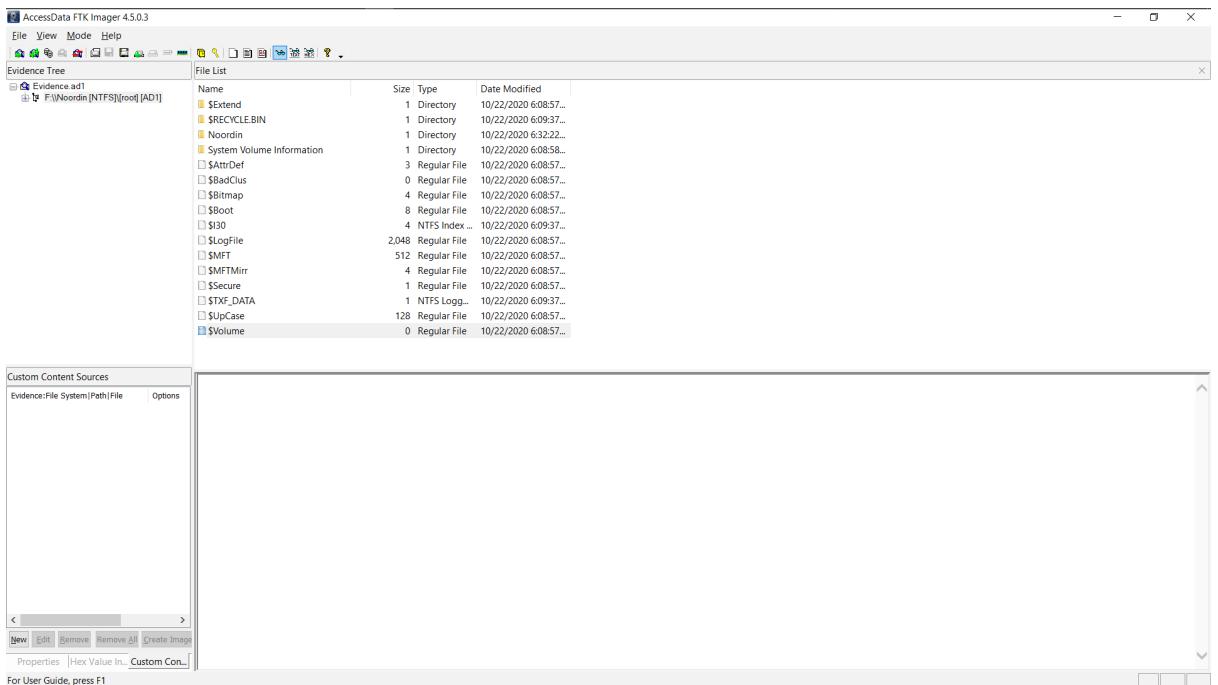
## Solusi

Pada soal ini diberikan dua buah file:

- Evidence.ad1
- cek\_integritas\_data.txt

Kedua file ini adalah file yg didapatkan sesudah membuat *disk image* dengan program FTK Imager.

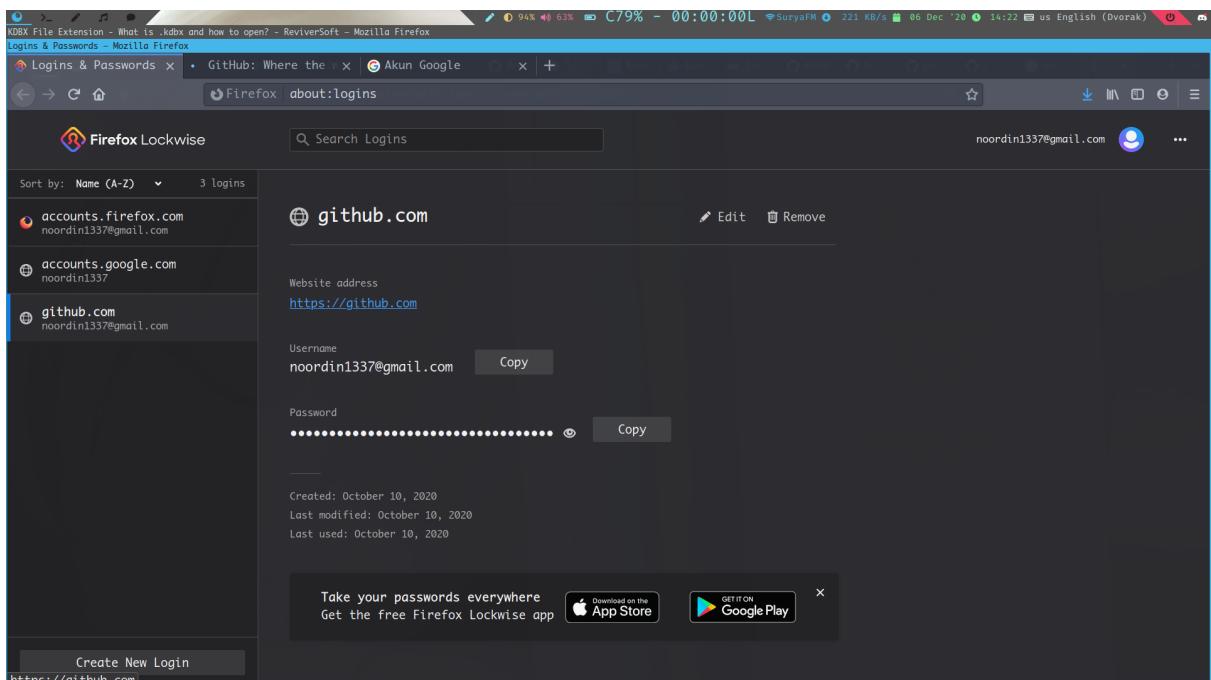
1. Buka file Evidence.ad1 pada FTK Imager.



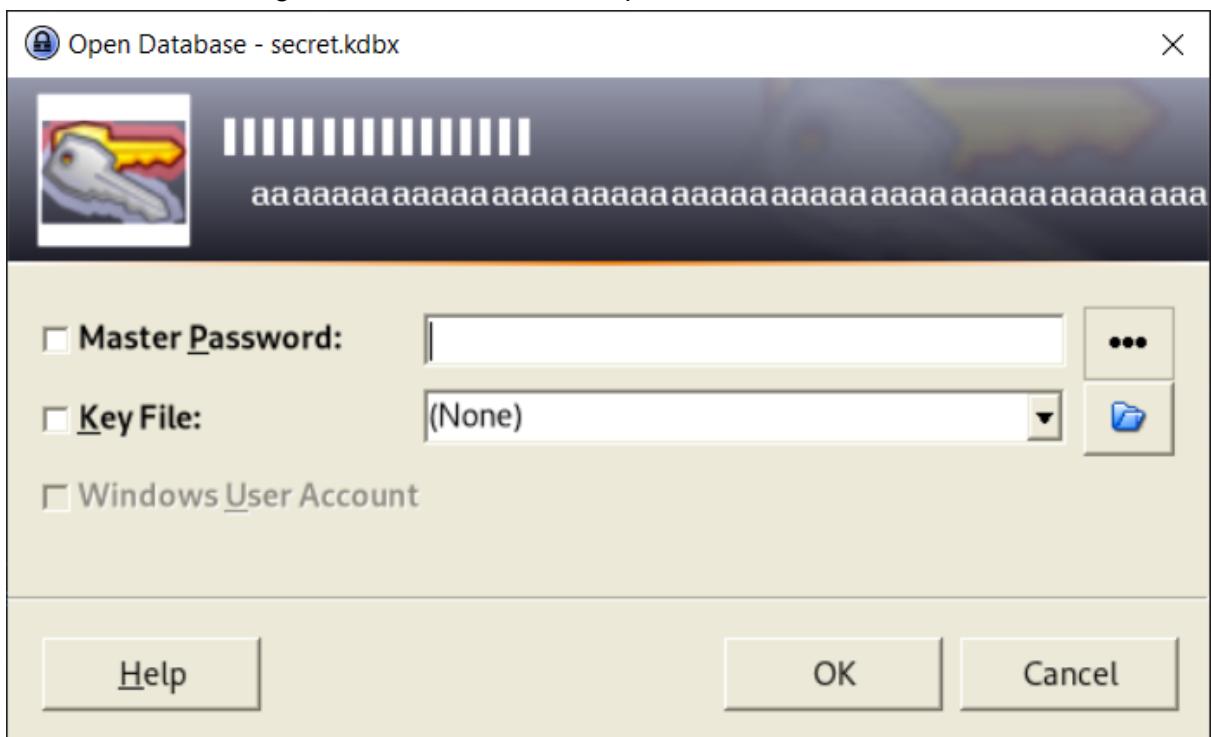
2. Lalu, pilih semua file dan pilih menu File -> Export Files.

File Path	File Name	File Type	File Size	File Date
Arch	\$Extend	File	4.8K	Dec 6 15:14
Arch	\$RECYCLE.BIN	File	4.8K	Dec 6 15:14
Arch	Noordin	File	4.8K	Dec 6 15:14
Arch	System Volume Information	File	4.8K	Dec 6 15:14
Arch	\$AttrDef	File	4.8K	Oct 22 13:08
Arch	\$BadClus	File	4.8K	Oct 22 13:08
Arch	\$Bitmap	File	4.8K	Oct 22 13:08
Arch	\$Boot	File	4.8K	Oct 22 13:08
Arch	\$I30	File	4.8K	Oct 22 13:08
Arch	\$LogFile	File	4.8K	Oct 22 13:08
Arch	\$MFT	File	4.8K	Oct 22 13:08
Arch	\$MFTMirr	File	4.8K	Oct 22 13:08
Arch	\$Secure	File	4.8K	Oct 22 13:08
Arch	\$TxF_DATA	File	4.8K	Oct 22 13:08
Arch	\$UpCase	File	4.8K	Oct 22 13:08
Arch	\$Volume	File	4.8K	Oct 22 13:08
Arch	\$Deleted	File	4.8K	Dec 6 15:14
Arch	\$ObjMetadata	File	4.8K	Dec 6 15:14
Arch	\$ObjId	File	4.8K	Oct 22 13:08
Arch	\$Quota	File	4.8K	Oct 22 13:08
Arch	\$Reparse	File	4.8K	Oct 22 13:08

3. Untuk mendapatkan password kedua dan ketiga, navigasi ke direktori Noordin/AppData/Mozilla/Firefox/Profiles, kemudian salin isi direktori tersebut ke direktori Profiles Firefox di komputer pemain CTF.
4. Buka Profile milik Noordin di Firefox pemain CTF, didapatkan:



5. Jika password di-show, isinya adalah:
  - w3akest\_link\_of\_53cNet\_
  - 15\_h00man\_so\_u\_c4nt\_trust\_em\_4ll! }
6. Selanjutnya, dilakukan digging lagi, lalu ditemukan file Noordin/privat/secret.kbdx . Setelah ditelusuri lagi, itu merupakan file database untuk password manager keepass.
7. Ketika file tersebut ingin dibuka, diminta master password.



8. Setelah melakukan penggalian lagi, ditemukan screenshot chat WhatsApp yang isinya mengatakan master password-nya terdapat di file excel yang sudah dienkripsi.
9. Ketika file Excel dibuka, semuanya memang dienkripsi.

10. Pada folder \$RECYCLE.BIN ditemukan program encryptor dan decryptornya.

11. Kedua file-decompile dengan uncompyle.

```
joseph@Kratos-W:~/ctf/CTF_2020/Forensics/Secket_Anomaly/decrypt//dec.cpython-38.py

[+] dec.pynt          Dec  6 14:51           [+] dec.cpython-38.py          Dec  6 14:28      # uncompiled version 3.7.4
[+] Nurdin           Dec  6 15:14           [+] dec_password.txt        Dec  6 14:04      # Python bytecode 3.8 (cli)
[+] cev_integratis ~ Dec  6 15:02           [+] dec_username.txt        Dec  6 14:31      # Decompiled from: Python 2.7.18 (default, Sep  5 2020, 11:17:26)
[+] Evidence.ad1     Dec  6 15:02           [+] enc_password.txt        Dec  6 14:02      # [GCC 10.2.0]
[+] secket.zip       Dec  6 13:45           [+] enc_username.txt        Dec  6 14:05      # Warning: this version of Python has problems handling the Python 3 "byte" type in constants properly
[+] pass.py          Dec  6 14:37           [+] pass.py                Dec  6 14:37      # Embedded file name: ./dec.py
[+] Compiled at: 2020-10-17 16:21:28
[+] # Size of source mod 2**32: 623 bytes

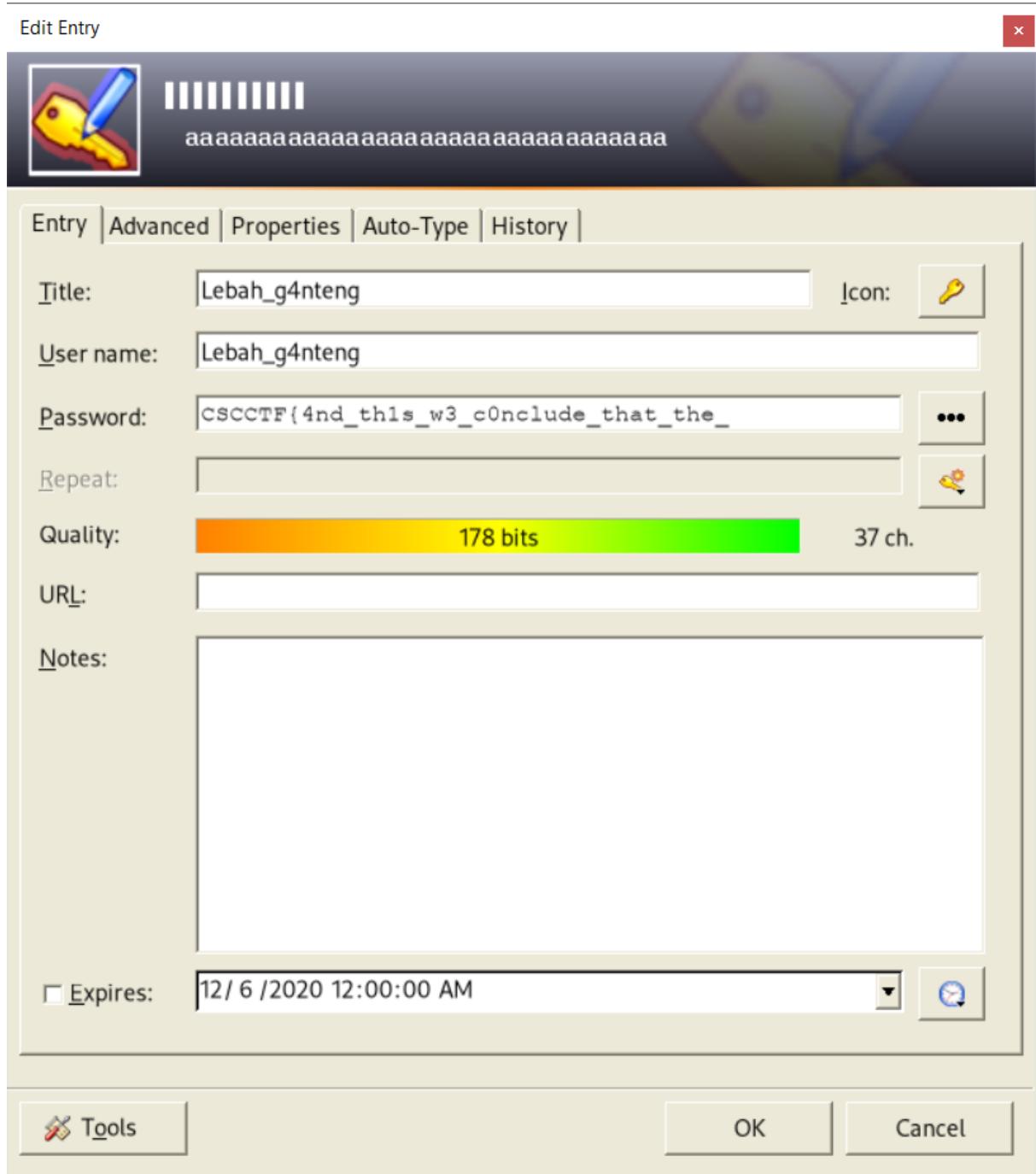
def decrypt(s):
    if len(bin(int(s, 16))[2:]) % 24 != 0:
        striped = '0' * (24 - len(bin(int(s, 16))[2:]) % 24) + bin(int(s, 16))[2:]
    else:
        striped = bin(int(s, 16))[2:]
    striped = [striped[i:i+24] for i in range(0, len(striped), 24)]
    for s in range(0, len(striped), 2):
        new = ''
        for i in range(len(striped[s])):
            new += str(int(striped[s][i]) ^ int(striped[s - 1][i]))
        else:
            striped[s] = new
    else:
        for s in range(0, len(striped), 2):
            striped[s] = striped[s][::-1]
        else:
            dec = ''
            for s in striped:
                dec += chr(int(s[0:8], 2)) + chr(int(s[8:16], 2)) + chr(int(s[16:24], 2))
        else:
            return dec

# with open('/home/wali/Desktop/enc_password.txt') as (x):
with open('enc_username.txt') as (x):
    for enc in x:
        enc = enc.strip()
        print(decrypt(enc))
# okay decompiling dec.cpython-38.pyc
```

12. lalu digunakan decryptor untuk mendapatkan username dan password sebenarnya pada excel.

13. Menurut deskripsi soal, username Noordin adalah Lebah\_g4nteng, Password-nya di Excel adalah 5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8.
  14. Lalu, digunakan password itu untuk keepass, didapatkan

The screenshot shows the KeePass application interface. The title bar reads "secret.kdbx - KeePass". The menu bar includes File, Group, Entry, Find, View, Tools, and Help. Below the menu is a toolbar with icons for New, Open, Save, Print, Find, Copy, Paste, and Lock. A search bar is also present. On the left, a tree view shows a single group named "secret" containing several entries: General, Windows, Network, Internet, eMail, Homebanking, and Recycle Bin. The main pane displays a table with columns: Title, User Name, Password, URL, and Notes. One entry is visible: "Lebah\_g4nteng Lebah\_g4nteng" with a masked password "\*\*\*\*\*". At the bottom, status bars show "0 of 1 selected" and "Ready".



Didapatkan:

CSCCTF{4nd\_th1s\_w3\_c0nclude\_that\_the\_w3akest\_link\_of\_53cNet\_15\_h00man\_so\_u\_c4nt\_trust\_em\_4ll!}

# Who's That?

## Soal

Aku baru saja menemukan bahwa pacarku selingkuh! Bantu aku mencari tahu siapa wanita selingkuhan pacarku itu! Wrap your hexadecimal output with the format CSCCTF{hexadecimal}. The hexadecimal should be all in CAPITAL LETTERS, the 0x heading is not needed!

Author: darmads

## Flag

CSCCTF{BECCA}

## Solusi

1. Diberikan sebuah file kode assembly:

```
.LC0:
    .string "%x\n"

main:
    push  rbp
    mov   rbp, rsp
    sub   rsp, 16
    mov   DWORD PTR [rbp-4], 415187
    mov   DWORD PTR [rbp-8], 0
    jmp   .L2

.L3:
    mov   edx, DWORD PTR [rbp-8]
    mov   eax, edx
    sal   eax, 2
    add   eax, edx
    add   eax, eax
    mov   edi, eax
    mov   edx, DWORD PTR [rbp-4]
    movsx  rax, edx
    imul  rax, rax, 1717986919
    shr   rax, 32
    sar   eax, 2
    mov   esi, edx
    sar   esi, 31
    sub   eax, esi
    mov   ecx, eax
    mov   eax, ecx
    sal   eax, 2
    add   eax, ecx
```

```
add    eax, eax
mov    ecx, edx
sub    ecx, eax
lea    eax, [rdi+rcx]
mov    DWORD PTR [rbp-8], eax
mov    eax, DWORD PTR [rbp-4]
movsx  rdx, eax
imul  rdx, rdx, 1717986919
shr    rdx, 32
sar    edx, 2
sar    eax, 31
mov    ecx, eax
mov    eax, edx
sub    eax, ecx
mov    DWORD PTR [rbp-4], eax
.L2:
cmp    DWORD PTR [rbp-4], 0
jg     .L3
mov    eax, DWORD PTR [rbp-8]
mov    esi, eax
mov    edi, OFFSET FLAT:.LC0
mov    eax, 0
call   printf
mov    eax, 0
leave
ret
```

2. Modifikasi kode assembly menjadi:

```
SECTION .data
f db '%x\n', 10

section .text
extern printf
global _start

_start:
push  rbp
mov   rbp, rsp
sub   rsp, 16
mov   DWORD [rbp-4], 415187
mov   DWORD [rbp-8], 0
jmp   .L2
.L3:
mov   edx, DWORD [rbp-8]
mov   eax, edx
sal   eax, 2
```

```
add    eax, edx
add    eax, eax
mov    edi, eax
mov    edx, DWORD [rbp-4]
movsx   rax, edx
imul   rax, rax, 1717986919
shr    rax, 32
sar    eax, 2
mov    esi, edx
sar    esi, 31
sub    eax, esi
mov    ecx, eax
mov    eax, ecx
sal    eax, 2
add    eax, ecx
add    eax, eax
mov    ecx, edx
sub    ecx, eax
lea    eax, [rdi+rcx]
mov    DWORD [rbp-8], eax
mov    eax, DWORD [rbp-4]
movsx   rdx, eax
imul   rdx, rdx, 1717986919
shr    rdx, 32
sar    edx, 2
sar    eax, 31
mov    ecx, eax
mov    eax, edx
sub    eax, ecx
mov    DWORD [rbp-4], eax
.L2:
cmp    DWORD [rbp-4], 0
jg    .L3
mov    eax, DWORD [rbp-8]
mov    esi, eax
mov    edi, f
mov    eax, 0
call   printf
mov    eax, 0
leave
ret
```

3. Dengan referensi

<https://montcs.bloomu.edu/~bobmon/Code/Asm.and.C/hello-asms.html>, Compile kode assembly menjadi object yang sudah dimodifikasi dengan command dan flag:

```
nasm -f elf64 -o girlfriend.o girlfriend.asm  
ld girlfriend.o -lc --dynamic-linker /lib/ld-2.32.so
```

4. Jalankan kode yang sudah dicompile

```
[0] [~/Downloads] % ./girlfriend  
becca\n  
zsh: segmentation fault (core dumped) ./girlfriend
```

5. Flag:

CSCCTF{BECCA}

## Welcome To CSCCTF

Diberikan sebuah file apk cscctf.apk. Karena soal ini reverse, maka insting pertama yang dilakukan adalah mencari decompiler. Sebelum didecompile, ada satu kunci, yaitu apk adalah file zip yang berisi asset dan kode. Maka, yang didecompile pasti ada di dalam apk tersebut. Buka file apk, cari file yang dapat didecompile. Didapat classes.dex dan classes2.dex. File-file tersebut dapat didecompile dengan jadx. Lalu, setelah didecompile, didapat pengecekan input ada di classes2.dex. Dengan z3solver, didapat kunci dari soal ini, yang constraintnya cukup banyak.

## Here We Go Again

Another good old menu in our cryptography section

Soal RSA dengan n, e, dan c. Dibuat code dalam python untuk solvenya

```
array= []  
  
flag=""  
  
for i in range(256):  
  
    m = pow(i,e,n)  
  
    array.append(m)  
  
  
lenCipher = len(cipher)  
  
  
for i in range(lenCipher):  
  
    for j in range(256):  
  
        if (cipher[i] == array[j]):  
  
            flag+=(chr(j))  
  
print(flag)
```

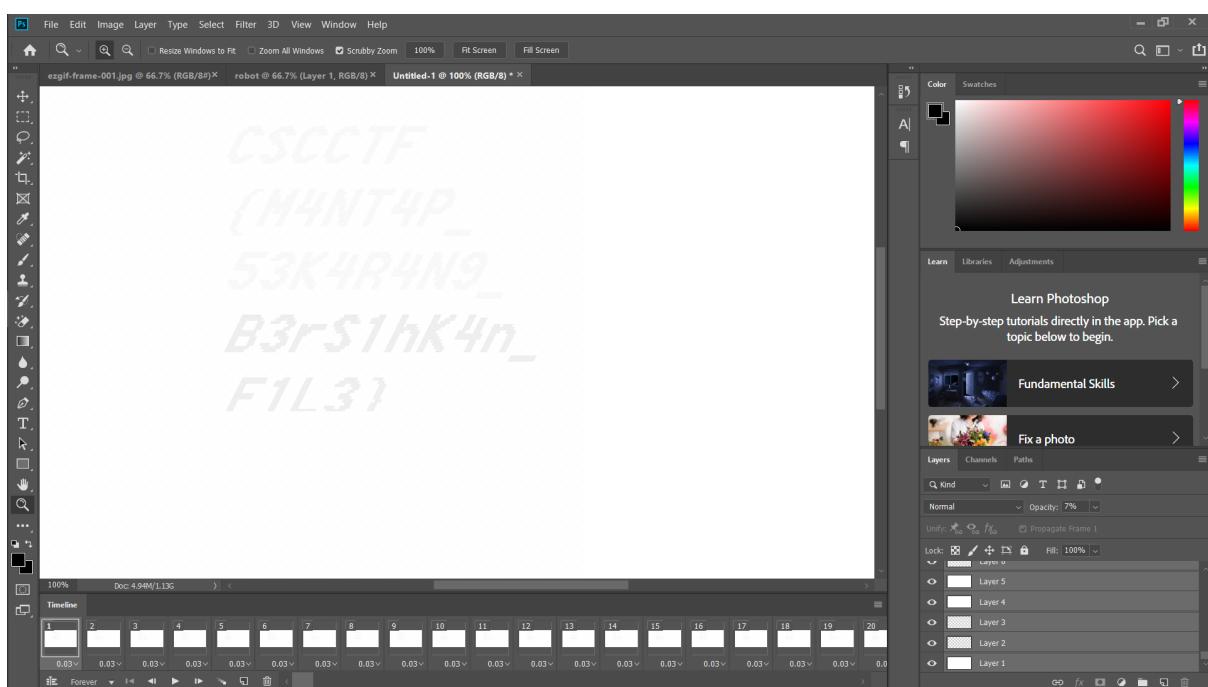
maka akan didapatkan flag CSCCTF{Rs4\_d3crYpt10n\_By\_3ncRypT10n}

## Aing, Robot

Sketch like Sonny <https://www.youtube.com/watch?v=Bs60aWyLrnI>

Author: Bigby

Download video yang telah diberikan, kemudian import video per frame ke dalam photoshop dan turunkan opacity pada semua layer hingga flag terlihat



flag: CSCCTF{M4NT4P\_53K4R4N9\_B3rS1hK4n\_F1L3}

## iHateDevelopers

### Soal

### Flag

CSCCTF{for\_those\_wh0\_rely\_on\_JS\_4lone}

## Solusi

Flag langsung muncul ketika web-page di-inspect.

```
<p id="the-secret" aria-hidden="true" hidden="">  
Flag is CSCCTF{for_those_wh0_rely_on_JS_4lone}</p>
```

## so-damn-smart

### Soal

I only wanna marry someone who's smarter than me - xomeone

Author: ArkAngels

Flag ada di /

### Flag

CSCCTF{you\_are\_smarter\_than\_MEH!}

## Solusi

1. Buka webpage, ada tampilan input sederhana
2. Tentukan tipe serangan yang dapat dilakukan, dalam kasus ini adalah SSTI karena setelah dicoba {7\*7} keluar 49. Setelah itu diketahui juga bahwa templating yang digunakan adalah Smarty (PHP) karena ada di nama soal, dan karena {7\*'7'} mengeluarkan 49.
3. Coba gunakan exec, filter tidak berhasil.
4. Coba gunakan backtick, bisa
5. Cari file flag, katanya di /, dengan input {php}echo `ls /{/php}
6. Lihat isi file flag, karena filenya terlalu panjang, gunakan shell expansion, {php}echo `cat /fl\*`{/php}
7. Dapat flag :D