

PicoCTF 2022

Ahmad Ryan Faizal

basic-file-exploit

The program provided allows you to write to a file and read what you wrote from it.

Try playing around with it and see if you can break it!

Connect to the program with netcat:

```
$ nc saturn.picoctf.net 52681
```

The program's source code with the flag redacted can be downloaded [here](#).

Sebelum saya connect ke programnya saya meliat source codenya dulu dan di source codenya saya menemukan suatu baris kode yang mencurigakan yaitu:

```
}
if ((entry_number = strtol(entry, NULL, 10)) == 0) {
    puts(flag);
    fseek(stdin, 0, SEEK_END);
    exit(0);
}

entry_number--;
strncpy(output, data[entry_number], input_lengths[entry_number]);
puts(output);
}

int main(int argc, char** argv) {
    ^G Help      ^O Write Out   ^W Where Is    ^K Cut         ^T Execut
    ^X Exit      ^R Read File   ^\ Replace     ^U Paste       ^J Justif
```

Di code dikatakan jika entry number = 0 maka akan dikeluarkan flag jadi langsung saja saya mencoba connect ke program dan langsung menggunakan cara seperti analisa saya diatas.

```
(kali@Yaan)-[~/LKS/Pico/2k22/file-exploit]
$ nc saturn.picoctf.net 52681

Hi, welcome to my echo chamber!
Type '1' to enter a phrase into our database
Type '2' to echo a phrase in our database
Type '3' to exit the program
1
1
Please enter your data:
Ryan
Ryan
Please enter the length of your data:
4
4
Your entry number is: 1
Write successful, would you like to do anything else?
2
2
Please enter the entry number of your data:
0
0
picoCTF{M4K3_5UR3_70_CH3CK_Y0UR_1NPU75_E0394EC0}
```

Flag: picoCTF{M4K3_5UR3_70_CH3CK_Y0UR_1NPU75_E0394EC0}

Enhance!

Download this image file and find the flag.

- [Download image file](#)

Disediakan image jika dibuka image tersebut didalamnya cuman ada gambar bulat begitu saja. Lalu saya mencoba melakukan strings ke image tersebut dan ada beberapa baris yang membuat saya curiga. Lalu saya menggunakan nano untuk melihat isi dari image file tersebut

```
y="132.08961"
style="font-size:0.00352781px;line-height:1.2
id="tspan3748">p </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.08942"
style="font-size:0.00352781px;line-height:1.2
id="tspan3754">i </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.09383"
style="font-size:0.00352781px;line-height:1.2
id="tspan3756">c </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.09824"
style="font-size:0.00352781px;line-height:1.2
id="tspan3758">o </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.10265"
style="font-size:0.00352781px;line-height:1.2
id="tspan3760">C </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.10706"
style="font-size:0.00352781px;line-height:1.2
id="tspan3762">T </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.11147"
style="font-size:0.00352781px;line-height:1.2
id="tspan3764">F { 3 n h 4 n </tspan><tspan
sodipodi:role="line"
x="107.43014"
y="132.11588"
style="font-size:0.00352781px;line-height:1.2
id="tspan3752">c 3 d _ 2 4 3 7 4 6 7 5 }</tsp
```

dan benar saja ada flag yang diselipkan di code

Flag: picoCTF{3nh4nc3d_24374675}

file-run1

A program has been provided to you, what happens if you try to run it on the command line?

Download the program [here](#).

Langsung saya menjalankan strings untuk melihat isi file tersebut dan benar saja ada flag didalamnya

```
(kali@Yaan)~/LKS/Pico/2k22/file-run1
$ strings run
/lib64/ld-linux-x86-64.so.2
qB7[0e6
libc.so.6
printf
__cxa_finalize
__libc_start_main
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
u+UH
[]A\A]A^A_
picoCTF{U51N6_Y0Ur_F1r57_F113_e5559d46}
```

picoCTF{U51N6_Y0Ur_F1r57_F113_e5559d46}

file-run2

Another program, but this time, it seems to want some input. What happens if you try to run it on the command line with input "Hello!"?

Download the program [here](#).

Saya ulangi lagi dengan cara di atas karena di kategori dan soal yang lumayan sama

```
(kali@Yaan)~/LKS/Pico/2k22/file-run2
$ strings run
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
printf
__cxa_finalize
strcmp
__libc_start_main
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
u+UH
[]A\A]A^A_
picoCTF{F1r57_4rgum3n7_96f2195f}
Run this file with only one argument.
Hello!
```

picoCTF{F1r57_4rgum3n7_96f2195f}

Lookey Here

Attackers have hidden information in a very large mass of data in the past, maybe they are still doing it.

Download the data [here](#).

Diberikan file berupa txt setelah saya mencoba membukanya keluarlah beberapa kalimat yang sangatlah bannyak lalu saya mencoba mengkobinasikan cat dan flag untuk mencari flag

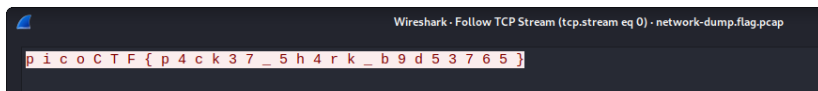
```
(kali@Yaan)~[~/LKS/Pico/2k22/Lookey]
$ cat anthem.flag.txt | grep pico
we think that the men of picoCTF{gr3p_15_@w3s0m3_58f5c024}
```

Packets Primer

Download the packet capture file and use packet analysis software to find the flag.

- [Download packet capture](#)

Diberikan packet capture setelah membuka langsung saja saya mencoba melakukan follow TCP stream

A screenshot of the Wireshark interface showing a TCP stream. The packet list on the left shows a packet of type 'HTTP' with status '200 OK'. The packet details pane on the right shows the 'Hypertext Transfer Protocol' section with a 'Content-Type' of 'text/html'. The packet bytes pane at the bottom shows the raw data of the packet, which is a flag: 'picoCTF{p4ck37_5h4rk_b9d53765}'. The flag is highlighted in red.

```
Wireshark - Follow TCP Stream (tcp.stream eq 0) - network-dump.flag.pcap
picoCTF{p4ck37_5h4rk_b9d53765}
```

picoCTF{p4ck37_5h4rk_b9d53765}

Redaction Gone Wrong

Now you DON'T see me.

This [report](#) has some critical data in it, some of which have been redacted correctly, while some were not. Can you find an important key that was not redacted properly?

Diberikan file pdf dan jika dibuka maka akan terlihat

Financial Report for ABC Labs, Kigali,

[REDACTED] - Just painted over in MS

[REDACTED]

Cost Benefit Analysis

Credit Debit

[REDACTED]

Expenses from the [REDACTED]

[REDACTED]

Langsung saja saya mencoba melihat apa yang ditutupi oleh garis hitam tersebut

Breakdown - Just painted over in MS

[REDACTED]

Cost Benefit Analysis

Credit Debit

This is not the flag, keep looking

Expenses from the [REDACTED]

picoCTF{C4n_Y0u_S33_m3_fully}

Redacted document.

picoCTF{C4n_Y0u_S33_m3_fully}

Eavesdrop

Download this packet capture and find the flag.

- [Download packet capture](#)

Disini langsung saya mencoba melakukan follow tcp stream dan ada muncul percakapan ini

```
Wireshark - Follow TCP Stream (tcp.stream eq 0) - capture.flag.pcap
Hey, how do you decrypt this file again?
You're serious?
Yeah, I'm serious
*sigh* openssl des3 -d -salt -in file.des3 -out file.txt -k supersecretpassword123
Ok, great, thanks.
Let's use Discord next time, it's more secure.
C'mon, no one knows we use this program like this!
Whatever.
Hey.
Yeah?
Could you transfer the file to me again?
Oh great. Ok, over 9002?
Yeah, listening.
Sent it.
Got it.
You're unbelievable
```

diatas ada command `openssl des3 -d -salt -in file.des3 -out file.txt -k supersecretpassword123` yang berarti ada command terjadi jika kita analisa maka des3 adalah nama file -salt berupa indikator -in berarti didalam sedangkan -out mungkin adalah hasil ekstrak ataupun file yang berada didalam. Disini langsung saya mencari file des3 ini sampai pada stream ke 2 ada seperti yang mencurigakan

```
Wireshark - Follow TCP Stream (tcp.stream eq 2) -
Salted____.0.G....^..GZ LbvBJ5eYm...R...@.M.U..
```

Disini saya berasumsi bahwa teks yang tidak jelas ini adalah file yang dikirimkan jadi saya berinisiatif mengexport packet byte ini dengan nama flag.txt

Setelah tereksport langsung saja saya mengganti nama file dengan des3 karena seperti yang ada pada command diatas lalu saya mencoba menjalankan file tersebut dan yang keluar adalah:

```
(kali@Yaan)-[~/LKS/Pico/2k22/Eavesdrop]
$ mv flag.txt file.des3

(kali@Yaan)-[~/LKS/Pico/2k22/Eavesdrop]
$ openssl des3 -d -salt -in file.des3 -out file.txt -k supersecretpassword123
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

(kali@Yaan)-[~/LKS/Pico/2k22/Eavesdrop]
$ cat file.
cat: file.: No such file or directory

(kali@Yaan)-[~/LKS/Pico/2k22/Eavesdrop]
$ cat file.txt
picoCTF{nc_73115_411_dd54ab67}
```

picoCTF{nc_73115_411_dd54ab67}

St3g0

Download this image and find the flag.

- [Download image](#)

Saya buka image yang disediakan tapi setelah dibuka yang keluar cuman lambang picoCTF disini karena judul soal Stego maka saya berasumsi ini adalah steganography lalu saya mencari tools yang berguna untuk Steganography dan ketemulah zsteg yang bisa mendeteksi data tersembunyi. Jadi langsung saja saya install dan mencoba menjalankan programnya

```
(kali@Yaan)-[~/LKS/Pico/2k22/St3g0]
$ zsteg pico.flag.png
b1,rgb,lsb,xy    .. text: "picoCTF{7h3r3_15_n0_5p00n_a1062667}$t3g0"
b1,abgr,lsb,xy   .. text: "E2A5q4E%uSA"
b2,b,lsb,xy      .. text: "AAPAAQTAAA"
b2,b,msb,xy      .. text: "HWUUUUUU"
b2,a,lsb,xy      .. file: Matlab v4 mat-file (little endian) >\004<\305P, numeric, rows 0, columns 0
b2,a,msb,xy      .. file: Matlab v4 mat-file (little endian) | <\243, numeric, rows 0, columns 0
b3,r,lsb,xy      .. file: gfxboot compiled html help file
b4,r,lsb,xy      .. file: Targa image data (16-273) 65536 x 4097 x 1 +4352 +4369 - 1-bit alpha - right "\021\
020\001\001\021\021\001\001\021\021\001"
b4,g,lsb,xy      .. file: 0420 Alliant virtual executable not stripped
b4,b,lsb,xy      .. file: Targa image data - Map 272 x 17 x 16 +257 +272 - 1-bit alpha "\020\001\021\001\021\
020\020\001\020\001\020\001"
b4,bgr,lsb,xy    .. file: Targa image data - Map 273 x 272 x 16 +1 +4113 - 1-bit alpha "\020\001\001\001"
b4,rgba,lsb,xy   .. file: Novell LANalyzer capture file
b4,rgba,msb,xy   .. file: Applesoft BASIC program data, first line number 8
b4,abgr,lsb,xy   .. file: Novell LANalyzer capture file
```

picoCTF{7h3r3_15_n0_5p00n_a1062667}\$t3g0

Morse Code

Morse code is well known. Can you decrypt this?

Download the file [here](#).

Wrap your answer with picoCTF{}, put underscores in place of pauses, and use all lowercase.

Diberikan file audio. Audio tersebut adalah morse code jadi saya menggunakan [decoder](#) ini untuk mengetahui apa isinya.

picoCTF{WH47_H37H_9oD_W2oU9H7}

patchme.py

Can you get the flag?

Run this [Python program](#) in the same directory as this [encrypted flag](#).

Diberikan sebuah program ketika dibuka program itu membutuhkan password jadi saya mencoba melihat kedalam program tersebut dan menemukan baris untuk input pw tersebut

```
def level_1_pw_check():
    user_pw = input("Please enter correct password for flag: ")
    if( user_pw == "ak98" + \
        "--90" + \
        "adfjhgj321" + \
        "sleuth9000"):
        print("Welcome back... your flag, user:")
        decryption = str_xor(flag_enc.decode(), "utilitarian")
        print(decryption)
        return
    print("That password is incorrect")

level_1_pw_check()
```

Jadi disini saya mengganti passwordnya dengan password yang saya tentukan

```
def level_1_pw_check():
    user_pw = input("Please enter correct password for flag: ")
    if( user_pw == "passwordpalingsusahdidunia"):
        print("Welcome back... your flag, user:")
        decryption = str_xor(flag_enc.decode(), "utilitarian")
        print(decryption)
        return
    print("That password is incorrect")
```

```
(kali@Yaan)~/LKS/Pico/2k22/patchme
$ nano patchme.flag.py

(kali@Yaan)~/LKS/Pico/2k22/patchme
$ python3 patchme.flag.py
Please enter correct password for flag: passwordpalingsusahdidunia
Welcome back... your flag, user:
picoCTF{p47ch1ng_l1f3_h4ck_f01eabfa}

(kali@Yaan)~/LKS/Pico/2k22/patchme
$
```

picoCTF{p47ch1ng_l1f3_h4ck_f01eabfa}

Sleuthkit Intro

Download the disk image and use `mmls` on it to find the size of the Linux partition. Connect to the remote checker service to check your answer and get the flag.

- [Download disk image](#)
- Access checker program: `nc saturn.picoctf.net 52279`

Diberikan disk dan suatu program untuk menghubungkan kita dengan flag saat saya mencoba program tersebut ditanyakan 'length in sectors' di disk yang disediakan disini langsung saja saya menggunakan `mmls` untuk cek disk tersebut

```
(kali@Yaan)-[~/LKS/Pico/2k22/Sleuthkit_Intro]
$ gzip -d disk.img.gz

(kali@Yaan)-[~/LKS/Pico/2k22/Sleuthkit_Intro]
$ ls
disk.img

(kali@Yaan)-[~/LKS/Pico/2k22/Sleuthkit_Intro]
$ mmls disk.img
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

   Slot   Start      End          Length      Description
000:  Meta   0000000000    0000000000    0000000001    Primary Table (#0)
001:  _____ 0000000000    0000002047    0000002048    Unallocated
002:  000:000 0000002048    0000204799    0000202752    Linux (0x83)

(kali@Yaan)-[~/LKS/Pico/2k22/Sleuthkit_Intro]
$ nc saturn.picoctf.net 52279
What is the size of the Linux partition in the given disk image?
Length in sectors: 202752
202752
Great work!
picoCTF{mm15_f7w!}
```

picoCTF{mm15_f7w!}

Substitution

A message has come in but it seems to be all scrambled. Luckily it seems to have the key at the beginning. Can you crack this substitution cipher?

Download the message [here](#).

Diberikan message yang acak seperti judulnya disini adalah substitution jadi saya mencari tool untuk memecahkan message ini dan ternyata metode yang digunakan adalah substitution cipher saya menggunakan [tool](#) ini di link ini <https://www.guballa.de/substitution-solver> karena tool ini adalah yang paling efisien

```
taking all things into consideration, I could hardly blame Jupit  
for his opinion  
respecting it.
```

```
The flag is: picoCTF{5UB5717U710N_3V0LU710N_03055505}
```

picoCTF{5UB5717U710N_3V0LU710N_03055505}

Substitution 1

A second message has come in the mail, and it seems almost identical to the first one. Maybe the same thing will work again.

Download the message [here](#).

Langkahnya sama seperti diatas dan menggunakan tool yang sama juga

```
taking all things into consideration, I could hardly blame Jupit  
for his opinion  
respecting it.
```

```
The flag is: picoCTF{5UB5717U710N_3V0LU710N_03055505}
```

picoCTF{FR3JU3NCY_4774CK5_4R3_C001_7AA384BC}

Substitution 2

It seems that another encrypted message has been intercepted. The encryptor seems to have learned their lesson though and now there isn't any punctuation! Can you still crack the cipher?

Download the message [here](#).

Langkah kali ini juga sama

picoCTF{N6R4M_4N41Y515_15_73D10U5_702F03FC}

```
velyorientedhighschoolcomputersecuritycompetitionthatseekstogenera  
teinterestincomputerscienceamonghighschoolersteachingthemenoughabo  
utcomputersecuritytopiquetheircuriositymotivatingthemtoexploreonth  
eirownandenablingthemtobetterdefendtheirmachinestheflagispicoCTF{N  
6R4M_4N41Y515_15_73D10U5_702F03FC}
```

Vigenere

Can you decrypt this message?

Decrypt this [message](#) using this key "CYLAB".

Disediakan pesan dan juga sebuah key seperti di judul soal 'vigenere' berarti pesan tersebut telah di enkripsi menggunakan vigenere cipher menggunakan key CYLAB. Saya menggunakan [tool ini](#) untuk mendecode pesan tersebut.

The screenshot shows an online Vigenere cipher decoder tool. The central panel has a 'Vigenère cipher' dropdown menu. Below it, there are four sections: 'VARIANT' with a dropdown set to 'Variant Beaufort cipher', 'KEY' with the text 'CYLAB', 'KEY MODE' with a dropdown set to 'Repeat', and 'ALPHABET' with the text 'abcdefghijklmnopqrstuvwxyz'. The left side panel has a text input field containing 'Q3_G1G303T3_A1AH'. The right side panel has a text output field containing 'picoCTF{D0NT_US3_V1G3N3R3_C1PH3R_d85729g7}'.

picoCTF{D0NT_US3_V1G3N3R3_C1PH3R_d85729g7}

Side Channel (400 points)

There's something fishy about this PIN-code checker, can you figure out the PIN and get the flag?

Download the PIN checker program here [pin_checker](#)

Once you've figured out the PIN (and gotten the checker program to accept it), connect to the master server using `nc saturn.picocftf.net 55824` and provide it the PIN to get your flag.

Soal kali ini masuk ke forensic tetapi entah kenapa tidak terasa seperti forensic.

Diberikan suatu program untuk check pin apakah benar atau salah seperti di hint dan judul soal ada yang namanya 'time side channel attack'. Setelah saya membaca hal ini simpelnya time side channel attack adalah suatu metode untuk menembus sesuatu menggunakan pengukuran waktu.

Kembali ke soal kita disediakan program untuk connect jika kita connect maka akan ditanyakan pin. Di soal disediakan program pin yang bisa didownload dengan program ini kita bisa mengexploit program ini. Disini saya membuat suatu script shell simpel:

```
GNU nano 6.4
#!/bin/bash
for x in {0..9}
do
    pin="{x}00000000"
    echo $pin
    time ./pin_checker <<< $pin
done
```

dan tidak lupa mengganti p 77ermission dari program pin tadi dengan chmod7 serta program yang dibuat dengan +x agar bisa di eksekusi.

```
(kali㉿kali)-[~/LKS/TimingAttack]
$ nano penembus_pin
(kali㉿kali)-[~/LKS/TimingAttack]
$ chmod 777 pin_checker
(kali㉿kali)-[~/LKS/TimingAttack]
$ chmod +x penembus_pin
```

setelah program tersebut dijalankan kita mencari baris pin yang paling lama dianalisis oleh program cek. Disini kasusnya adalah 40000000

```
30000000
Please enter your 8-digit PIN code:
8
Checking PIN ...
Access denied.

real    0m0.129s
user    0m0.128s
sys     0m0.000s

40000000
Please enter your 8-digit PIN code:
8
Checking PIN ...
Access denied.

real    0m0.257s
user    0m0.256s
sys     0m0.000s

50000000
```

Karena program tadi tidak sepenuhnya otomatis jadi kita harus menggeser 'loop-nya' lagi dan menambahkan bilangan yang sudah kita dapat.

```
#!/bin/bash
for x in {0..9}
do
    pin="4${x}000000"
    echo $pin
    time ./pin_checker <<< $pin
done
```

Lalu kita ulangi lagi proses diatas sampai kita menemukan ke 8 pin tersebut.

```
Access denied.

real    0m0.961s
user    0m0.942s
sys     0m0.014s
48390513
Please enter your 8-digit PIN code:
8
Checking PIN...
Access granted. You may use your PIN to log into the master server.
```

```
real    0m1.067s
user    0m1.041s
sys     0m0.024s
```

```
48390514
Please enter your 8-digit PIN code:
8
Checking PIN...
Access denied.
```

dan pinnya adalah 48390513 sekarang kita tinggal connect ke program tersebut dan memasukkan pinnya

```
(kali㉿kali)-[~/LKS/TimingAttack]
$ nc saturn.picocftf.net 55824
Verifying that you are a human ...
Please enter the master PIN code:
48390513
Password correct. Here's your flag:
picoCTF{t1m1ng_4tt4ck_eb4d7efb}
```

picoCTF{t1m1ng_4tt4ck_eb4d7efb}

Operation Oni (300 points)

Description

Download this disk image, find the key and log into the remote machine.

Note: if you are using the webshell, download and extract the disk image into `/tmp` not your home directory.

- [Download disk image](#)
- Remote machine: `ssh -i key_file -p 62607 ctf-player@saturn.picoctf.net`

Seperti biasa karena ini berbentuk image saya menggunakan mmls untuk mengetahui detail dari disk tersebut di soal dijelaskan kita harus mencari ssh key yang berada didalam email tersebut jadi yang pertama kali saya lakukan adalah mencari keberadaan 'ssh' dalam disk ini

```
(kali㉿kali)-[~/LKS]
└─$ fls -r -o 206848 disk.img | grep .ssh
++ r/r 2147:      sshd
+++ l/l 54:      sshd
++ r/r 2148:      sshd
+ d/d 14:        ssh
++ r/r 15:        ssh_host_ed25519_key
++ r/r 16:        ssh_host_ed25519_key.pub
++ r/r 17:        ssh_host_ecdsa_key
++ r/r 18:        ssh_host_ecdsa_key.pub
++ r/r 19:        ssh_host_dsa_key
++ r/r 20:        ssh_host_dsa_key.pub
++ r/r 21:        ssh_host_rsa_key
++ r/r 22:        ssh_host_rsa_key.pub
++ r/r 2136:      ssh_config
++ r/r 2149:      sshd_config
++ r/r 2084:      ssh-keygen
++ r/- * 0:       ssh-copy-id
++ r/- * 0:       ssh-keyscan
++ r/- * 0:       ssh-pkcs11-helper
++ r/r 2140:      ssh-add
++ r/r 2145:      ssh
++ r/r 2144:      ssh-pkcs11-helper
++ r/r 2143:      ssh-keyscan
++ r/r 2142:      ssh-copy-id
++ r/r 2141:      ssh-agent
++ r/r 2150:      sshd
+++++ r/r 676:    sshd
++ d/d 3907:      ssh
+++ r/r 2152:      ssh-sk-helper
+++ r/r 2151:      ssh-pkcs11-helper
+ r/r 712:        setup-sshd
+ d/d 3916:       .ssh
```

Disini saya melihat beberapa ssh key saya mencoba semuanya dan hasilnya nihil karena masih diminta password ketika sudah connect ke sshnya

```
(kali㉿kali)-[~/LKS]
$ ssh -i yaan2.key -p 62607 ctf-player@saturn.picoc.tf.net
ctf-player@saturn.picoc.tf.net's password:

zsh: suspended ssh -i yaan2.key -p 62607 ctf-player@saturn.picoc.tf.net

(kali㉿kali)-[~/LKS]
$ ssh -i yaan.key -p 62607 ctf-player@saturn.picoc.tf.net
ctf-player@saturn.picoc.tf.net's password: █
```

Disini saya tertarik pada 2 file karena berbentuk directory dan juga posisi (inode) yang paling jauh yaitu sekitar 3900 sedangkan yang lain hanya 2000-an. Saya mencoba melihat kedalamnya dan di .ssh file ada file yang mencurigakan karena ada yang nama file yang depannya id dan belakangnya .pub kedua hal ini identik dengan ssh.

```
(kali㉿kali)-[~/LKS]
$ icat -o 206848 disk.img 3916
L
.♦
..)
id_ed25519* ♦id_ed25519.pub
          ♦♦♦/

(kali㉿kali)-[~/LKS]
$ icat -o 206848 disk.img 3907
C
.♦
ssh-sk-helper10ead6bd529bcc747d0372fcb1f0932b12feccd7T
sftp-server@5.apk.aaf7b6e9311b326086bba1413f8c2554daed95e4f2b
a320Hssh-pkcs11-helper
♦♦♦1
```

Jadi disini saya mencari info tentang 'id' ini dan juga menyimpan key tersebut

```
(kali㉿kali)-[~/LKS]
$ fls -r -o 206848 disk.img | grep id_ed
++ r/r 2345: id_ed25519
++ r/r 2346: id_ed25519.pub

(kali㉿kali)-[~/LKS]
$ icat -o 206848 disk.img 2345
-----BEGIN OPENSSH PRIVATE KEY-----
b3B1bnZaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAAAMwAAAAAtzc2gtZW
QyNTUxOQAAACBgrXe4bKNhOzkCLW0mk4zDMimW9RVZngX51Y8h3BmKLAAAAJgxpYKDMaWC
gwAAAAAtzc2gtZWQyNTUxOQAAACBgrXe4bKNhOzkCLW0mk4zDMimW9RVZngX51Y8h3BmKLA
AAAEICitu0F8DIjWxTp+KeMDvX1lQwYtUvP2SfSVOfMOChxYGCTd7hso2E70QItY6aTjMMY
KZb1FVmeBfnVjyHcGYosAAAADnJvb3RabG9jYWxob3N0AQIDBAUGBw==
-----END OPENSSH PRIVATE KEY-----

(kali㉿kali)-[~/LKS]
$ icat -o 206848 disk.img 2345 > yaan3.key
```


Disini saya harus merestart instance dari pico karena sudah kehabisan waktu.

```
(kali@kali)~[~/LKS]
$ chmod 600 yaan3.key

(kali@kali)~[~/LKS]
$ ssh -i yaan3.key -p 62607 ctf-player@saturn.picoctf.net
ssh: connect to host saturn.picoctf.net port 62607: Connection refused

(kali@kali)~[~/LKS]
$ ssh -i yaan3.key -p 58939 ctf-player@saturn.picoctf.net
The authenticity of host '[saturn.picoctf.net]:58939 ([18.217.86.78]:58939)' can't be established.
ED25519 key fingerprint is SHA256:5gIm/EJ9bYnoH4qed83W5HXLfN1D055849f6Lze0lx8.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[saturn.picoctf.net]:58939' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-1025-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

ctf-player@challenge:~$ ls
flag.txt
ctf-player@challenge:~$ cat flag.txt
picoCTF{k3y_5l3u7h_af277f77}ctf-player@challenge:~$
```

picoCTF{k3y_5l3u7h_af277f77}

Operation Orchid (400 points)

Diberikan disk seperti soal sebelumnya disini langsung saja saya melihat ke dalam isi disk tersebut. Didalamnya banyak sekali file yang bernama 'orphan' jadi disini saya mengecualikan file orphan tersebut dengan grep.

"fls -r -o 411648 disk.flag.img | grep -v Orphan"

```
+ r/r 1875:      .ash_history
+ r/r * 1876(realloc):  flag.txt
+ r/r 1782:      flag.txt.enc
d/d 473:        run
d/d 475:        srv
d/d 476:        sys
d/d 2041:       swap

(kali@Yaan)~[~/LKS/Pico/2k22/Operation_Orchid]
$ icat -o 411648 disk.flag.img 1782
Salted 0金 0eoz2>@++++SSgk(r)}+}+f+zZ7+ +++$+%


```

Disini ada file flag.txt.enc ketika dilihat kedalamnya flag ini sudah dienkripsi,

Diatas file flag tersebut saya melihat sesuatu yang mencurigakan yaitu .ash_history saya kira ini adalah suatu key untuk decode flag diatas jadi langsung saya melihat kedalam isi ash history ini dan tidak lupa juga saya mengeluarkan flag tersebut dari disk.

Ketika dilihat kedalam file history ternyata ini adalah history dari bagaimana caranya mereka mengenkripsi flag tersebut yaitu menggunakan OpenSSL dengan tambahan Salt

```
(kali@Yaan)-[~/LKS/Pico/2k22/Operation_Orchid]
$ icat -o 411648 disk.flag.img 1875
touch flag.txt
nano flag.txt
apk get nano
apk --help
apk add nano
nano flag.txt
openssl
openssl aes256 -salt -in flag.txt -out flag.txt.enc -k unbreakablepassword1234567
shred -u flag.txt
ls -al
halt
```

Jadi simpel saja kita ulangi command yang telah dieksekusi tetapi menambahkan -d didalamnya karena kita ingin mendecode teks tersebut

```
(kali@Yaan)-[~/LKS/Pico/2k22/Operation_Orchid]
$ openssl aes256 -d -salt -in flag.txt.enc -out iniflag.txt -k unbreakablepassword1234567
** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
bad decrypt
80EB96E95F7F0000:error:1C800064:Provider routines:ossl_cipher_unpadblock:bad decrypt:../providers/implementation
s/ciphers/ciphercommon_block.c:124:

(kali@Yaan)-[~/LKS/Pico/2k22/Operation_Orchid]
$ cat iniflag.txt
picoCTF{h4un71ng_p457_5113beab}
```

Sleuthkit Apprentince (200 Points)

Diberi file disk lagi seperti biasa saya melihat kedalam disk tersebut dan saat saya melihat kedalam disk tersebut ada banyak file orphan lagi jadi saya keculikan dan ada file yang berhubungan dengan flag.

```
+ r/r 2363:      .ash_history
+ d/d 3981:      my_folder
++ r/r * 2082(realloc): flag.txt
++ r/r 2371:      flag.uni.txt
d/d 1996:        run
d/d 1997:        srv
d/d 1998:        sys
d/d 2358:        swap

(kali@Yaan)-[~/LKS/Pico/2k22/Sleuthkit_Apprentince]
$ icat -o 360448 disk.flag.img 2371
picoCTF{by73_5urf3r_adac6cb4}
```

picoCTF{by73_5urf3r_adac6cb4}

File Type

This file was found among some files marked confidential but my pdf reader cannot read it, maybe yours can.

You can download the file from [here](#).

Diberikan file tetapi ketika diidentifikasi file tersebut bukanlah pdf melainkan shell.

```
(kali@Yaan)-[~/LKS/Pico/2k22/Type_File]
$ ./Flag.pdf
x - created lock directory _sh00046.
x - extracting flag (text)
./Flag.pdf: 119: uudecode: not found
restore of flag failed
flag: MD5 check failed
x - removed lock directory _sh00046.
```

Jika menggunakan kali linux kita perlu install uudecode dulu agar programnya bisa dijalankan karena entah kenapa di kali linux uudecode belum terinstall.

```
(kali@Yaan)-[~/LKS/Pico/2k22]
$ cd Type_File

(kali@Yaan)-[~/LKS/Pico/2k22/Type_File]
$ ./Flag.pdf
x - created lock directory _sh00046.
x - extracting flag (text)
x - removed lock directory _sh00046.

(kali@Yaan)-[~/LKS/Pico/2k22/Type_File]
$ ls
flag  Flag.pdf  _Flag.pdf.extracted
```

Nah sekarang kita bisa melihat file flag hasil ekstrak dari program tadi. Ternyata file berbentuk arsip jadi saya binwalk untuk mengekstrak file yang berada didalamnya

```
(kali@Yaan)-[~/LKS/Pico/2k22/Type_File]
$ binwalk -e flag
```

DECIMAL	HEXADECIMAL	DESCRIPTION
100	0x64	bzip2 compressed data, block size = 900k

Sebenarnya lanjutan dari ini adalah tinggal decompress saja tapi bentuk decompressnya dengan berbagai bentuk zip.

```

375 file 64
376 gzip -d 64
377 mv 64 64.gz
378 gzip -e 64.gz
379 gzip -d 64.gz
380 ls
381 file 64
382 lzip
383 binwalk 64
384 binwalk -e 64
385 ls
386 lzip
387 sudo apt install lzip
388 lzip -d 64
389 ls
390 file 64.out
391 unlz4
392 unlz4 64.out
393 unlz4 64.out flag
394 ls
395 file flag
396 rm 64.out
397 lzma -d flag
398 mv flag flag.lzma
399 lzma -d flag.lzma
400 ls
401 file flag
402 lzop
403 lzop -d flag
404 ls
405 mv flag flag.lzop
406 lzop -d flag.lzop
407 ls
408 rm flag.lzop
409 file flag
410 lzip -d flag.lz
411 lzip -d flag
412 ls
413 file flag.out
414 sudo apt install xz-utils
415 unxz flag.out
416 mv flag.out flag.xz

```

Setelah kita melewati proses extract yang sangatlah panjang kita akan diberikan satu file berbentuk text yang diencode dengan hex.

```

(kali@Yaan)-[~/Pico/2k22/Type_File/_flag.extracted]
$ ls
flag

(kali@Yaan)-[~/Pico/2k22/Type_File/_flag.extracted]
$ hex -d flag
picoCTF{f1len@m3_m@n1pul@t10n_f0r_0b2cur17y_79b01c26}

```

picoCTF{f1len@m3_m@n1pul@t10n_f0r_0b2cur17y_79b01c26}

