

PicoCTF 2021  
Ahmad Ryan Faizal

## Obedient Cat

*This file has a flag in plain sight (aka "in-the-clear"). [Download flag](#).*

Sesuai soal flag ada di plain sight berarti mudah saja langsung cat dan ya benar saja flagnya langsung keliatan

```
(kali㉿kali)-[~/LKS/Pico/2k21]
$ cat flag
picoCTF{s4n1ty_v3r1f13d_f28ac910}

(kali㉿kali)-[~/LKS/Pico/2k21]
$
```

picoCTF{next\_time\_I'll\_try\_2\_rounds\_of\_rot13\_aFxtzQWR}

## Mod26

*Cryptography can be easy, do you know what ROT13 is?*  
*cvpbPGS{arkg\_gvzr\_V'yy\_gel\_2\_ebhaqf\_bs\_ebg13\_nSkgmDJE}*

Sesuai soal ditanyakan rot13 maka langsung saja saya convert dengan web rot13

```
picoCTF{next_time_I'll_try_2_rounds_of_rot13_aFxtzQWR}
```

picoCTF{next\_time\_I'll\_try\_2\_rounds\_of\_rot13\_aFxtzQWR}

## Python Wrangling

Python scripts are invoked kind of like programs in the Terminal... Can you run [this Python script](#) using [this password](#) to get [the flag](#)?

Sesuai soal disuruh menjalankan program yang sudah disediakan lalu mendapatkan flag dengan menggunakan password yang disediakan juga

Pertama-tama saya liat apa password tersebut setelah mendapatkannya saya langsung menjalankan programnya ternyata memiliki dua opsi antara -e/-d lalu dimasukkan file yang ingin didapatkan yang disini kondisinya flag. Saya coba saja dua-duanya dan ya benar flagnya didapatkan

```
(kali@kali)-[~/LKS/Pico/2k21/Python_Wrangling]
$ cat pw.txt
6008014f6008014f6008014f6008014f

(kali@kali)-[~/LKS/Pico/2k21/Python_Wrangling]
$ python3 ende.py
Usage: ende.py (-e/-d) [file]

(kali@kali)-[~/LKS/Pico/2k21/Python_Wrangling]
$ python3 ende.py -e flag.txt.en
Please enter the password:6008014f6008014f6008014f6008014f
gAAAAABjKG2-c8JK-0hwB4Le3epVywofprtlzkyG8mP-eUKGC_CVh0h3Z5zf84bC2Yl9CP140PF2e_tlUL8qjMeLdrgvkjzsH8JTNqzYYirX3w
LfkFENPxt4G3NlsbzQorEU8lFHX7FtNHm7YkjlnOaeYKPH_AvVmyqvFNhzWTRXRbYDvL8I3oYVQ07rGJASMkYK0JZqymjWbRfQo0aa4xo2J7DvA
FLQKIBc08VT6rwBK-E077ePWrf-bHkv9nacuZUAuD0ztFM

(kali@kali)-[~/LKS/Pico/2k21/Python_Wrangling]
$ python3 ende.py -d flag.txt.en
Please enter the password:6008014f6008014f6008014f6008014f
picoCTF{4p0110_1n_7h3_h0us3_6008014f}
```

## Wave a flag

Can you invoke help flags for a tool or binary? [This program](#) has extraordinarily helpful  
Disediakan program 'warm' karena sudah kebiasaan saya sebelum menjalankan suatu  
information...

file/program selalu saya 'cat' ataupun 'nano' file tersebut dan setelah saya cat saya lihat flagnya langsung muncul.

```
*****H*****Q]*****FDUH*****f*****UH*****H*****H*****uH*****KH*****E*****H*****H*****5*****H*****uH*****i*****H*****E*****H*****H*****H*****=
**X*****DAWAVI*****AUATL*%F UH*-F SA*****I*****L)*H*****H*****H*****t 1*****L*****L*****D*****A*****H*****H*****9*****u*****H*****[A\A]A^A_df.*****H*****Hello us
er! Pass me a -h to learn what I can do!-hOh, help? I actually don't do much, but I do have this flag here: pic
oCTF{b1sculTs_4nd_gr4vy_d6969390}I don't know what '%s' means! I do know what -h means though!
<*****X*****0zRx
*****zRx
$8*****@F]
l
D|*****eB*B*E *B(*H0*H8*M@r8A0A(B B*B*****
*****o*****
*
```

picoCTF{b1sculTs\_4nd\_gr4vy\_d6969390}

## Alternative / Actual Way

Karena file tidak bisa dieksekusi dengan alasan “permission denied” maka saya ubah dengan permissionnya dengan menggunakan “chmod”

```
(kali@kali)-[~/LKS/Pico/2k21/Wave_a_flag]
$ chmod +x warm

(kali@kali)-[~/LKS/Pico/2k21/Wave_a_flag]
$ ./warm
Hello user! Pass me a -h to learn what I can do!

(kali@kali)-[~/LKS/Pico/2k21/Wave_a_flag]
$ ./warm -h
Oh, help? I actually don't do much, but I do have this flag here: picoCTF{b1scu1ts_4nd_gr4vy_d6969390}
```

picoCTF{b1scu1ts\_4nd\_gr4vy\_d6969390}

## information

*Files can always be changed in a secret way. Can you find the flag? [cat.jpg](#)*

Diberikan hint “look at the details of the file” tanpa basa-basi karena sebelumnya saya juga sudah biasa melihat details dari file langsung saja saya lihat menggunakan exiftool dan disana ada beberapa karakter yang mencurigakan yaitu “7a78f3d9cfb1ce42ab5a3aa30573d617” dan “cGljb0NURnt0aGVfbTN0YW RhdGFfMXNfbW9kaWZpZW R9”

```
$ exif cat.jpg
Corrupt data
The data provided does not follow the specification.
ExifLoader: The data supplied does not seem to contain EXIF data.

(kali@kali)-[~/LKS/Pico/2k21/information]
$ exiftool cat.jpg
ExifTool Version Number      : 12.44
File Name                    : cat.jpg
Directory                   : .
File Size                    : 878 kB
File Modification Date/Time   : 2021:03:15 14:24:46-04:00
File Access Date/Time        : 2022:09:19 09:46:04-04:00
File Inode Change Date/Time   : 2022:09:19 09:45:58-04:00
File Permissions              : -rw-r--r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                  : 1.02
Resolution Unit               : None
X Resolution                  : 1
Y Resolution                  : 1
Current IPTC Digest           : 7a78f3d9cfb1ce42ab5a3aa30573d617
Copyright Notice              : PicoCTF
Application Record Version    : 4
XMP Toolkit                   : Image::ExifTool 10.80
License                       : cGljb0NURnt0aGVfbTN0YW RhdGFfMXNfbW9kaWZpZW R9
Rights                        : PicoCTF
Image Width                   : 2560
Image Height                  : 1598
Encoding Process               : Baseline DCT, Huffman coding
Bits Per Sample                : 8
```

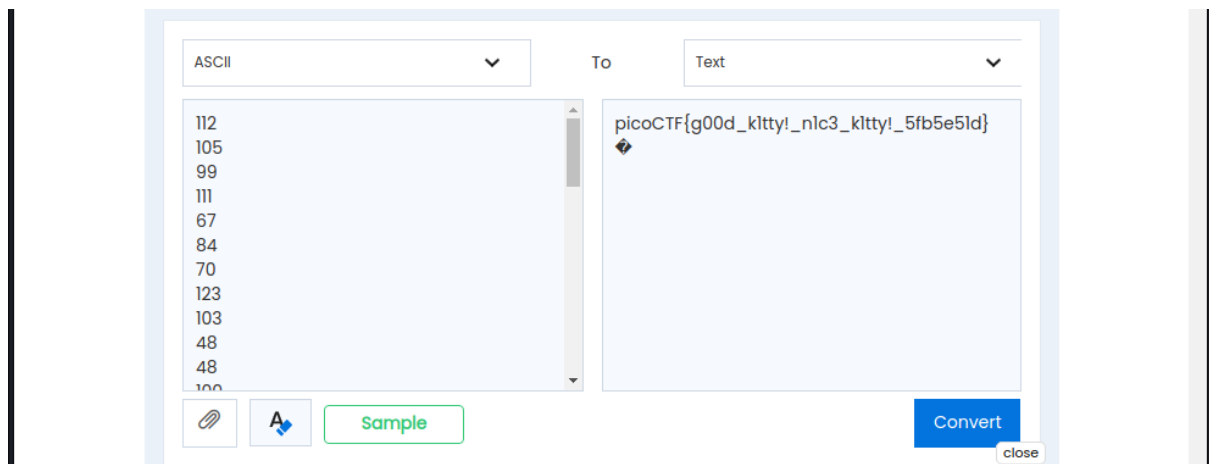
langsung coba saya decode di CyberChef dan CyberChef bisa mendeteksi bahwa salah satunya diencode menggunakan Base64 langsung saja saya decode dan keluarlah flagnya:

picoCTF{the\_m3tadata\_1s\_modified}

## Nice netcat..

There is a nice program that you can talk to by using this command in a shell: `$ nc mercury.picoctf.net 22342`, but it doesn't speak English...

Langsung saja saya coba netcat ke link yang sudah diberikan dan yang muncul adalah beberapa angka saya mengasumsikan itu adalah ASCII dan untuk memastikan kembali saya membuka hint yang diberikan dan yak benar saja ada hint yang berhubungan dengan ASCII maka saya convert menggunakan [website](#) ini



picoCTF{g00d\_k1tty!\_n1c3\_k1tty!\_5fb5e51d}

## Information

I wonder what this really is... `enc ''.join([chr((ord(flag[i]) << 8) + ord(flag[i + 1])) for i in range(0, len(flag), 2)])`

Disini saya sangatlah bingung dan tidak mempunyai clue apa yang harus dilakukan lalu saya coba membuka file flag yang terenkripsi tersebut dan muncullah

“濫捌宏規し形梃獠楮獠ズル樓潦彌彦レ一て魑”

Disini saya berasumsi ini adalah flag yang terenkripsi dengan suatu bahasa karena saya bingung itu bahasa apa saya kembali menggunakan cyberchef dengan fitur 'magic'-nya yang bisa detect dan decode berbagai bahasa dan ya setelah di dekripsi dengan berbagai bahasa langsung saja saya search pico dan ketemu flagnya

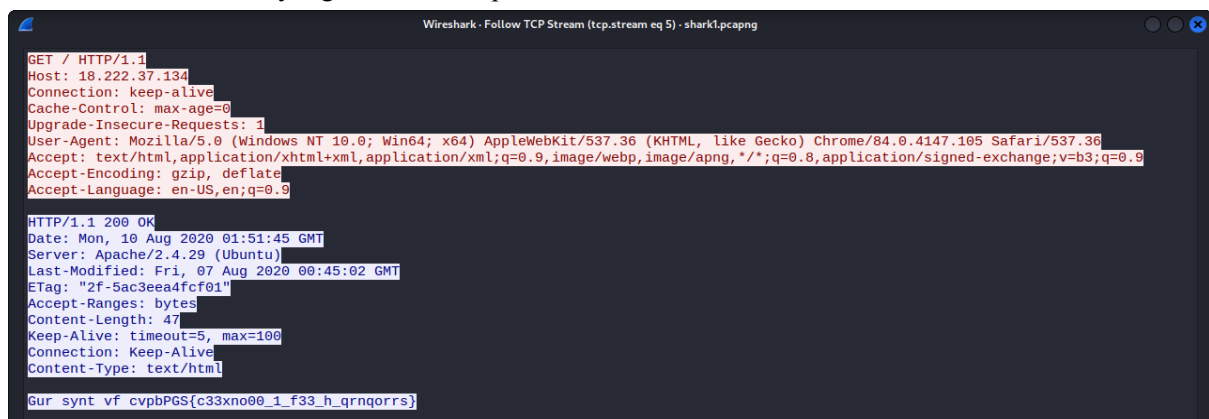


picoCTF{16\_bits\_inst34d\_of\_8\_e141a0f7}

## Wireshark doo dooo do doo...

Can you find the flag? [shark1.pcapng](#).

Setelah membuka filenya langsung saja saya follow TCP Stream sampai pada stream ke 5 saya menemukan suatu teks yang berbentuk seperti format ctf



Gur synt vf cvpbPGS{c33xno00\_1\_f33\_h\_qrnqorrs}

karena saya lumayan familiar dengan rot13 maka saya langsung tau ini rot13 dari “vf” karena vf jika di decrypt menggunakan rot13 adalah is. Maka langsung saja saya menggunakan web rot13 seperti biasa dan muncullah flag

The flag is picoCTF{p33kab00\_1\_s33\_u\_deadbeef}

## Tab, Tab, Attack

*Using tabcomplete in the Terminal will add years to your life, esp. when dealing with long rambling directory structures and filenames: [Addadshashanammu.zip](#)*

Saya mulai dengan mengunzip file tersebut didalam file tersebut ada beberapa lipat direktori jadi saya melihat sampai direktori terakhir dan ada file yang jika dijalankan akan memberikan flag

```
(kali㉿kali)-[~/Assurnabitashpi/Maelkashishi/Onnissiralis/Ularradallaku]
$ ./fang-of-haynekhtnamet
*ZAP!* picoCTF{l3v3l_up!_t4k3_4_r35t!_f3553887}

(kali㉿kali)-[~/Assurnabitashpi/Maelkashishi/Onnissiralis/Ularradallaku]
$
```

picoCTF{l3v3l\_up!\_t4k3\_4\_r35t!\_f3553887}

## Static ain't always noise

*Can you look at the data in this binary: [static](#)? This [BASH script](#) might help!*

Diberikan file static (?) dan juga script. Alih-alih menjalankan script yang diberikan saya melakukan strings pada file ‘static’ untuk melihat beberapa printable character dan ternyata ada flag didalamnya

```
AOATL
[]A\A]A^A_
Oh hai! Wait what? A flag? Yes, it's around here somewhere!
;*3$"
picoCTF{d15a5m_t34s3r_98d35619}
GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0
crtstuff.c
```

picoCTF{d15a5m\_t34s3r\_98d35619}

## Matryoshka doll

Matryoshka dolls are a set of wooden dolls of decreasing size placed one inside another.  
What's the final one? Image: [this](#)

Diadakan file image dan ada soal “one inside another” yang bisa diasumsikan ada hidden file dan benar saja ketika di binwalk ada hidden file yang keluar saya ulang proses binwalk ini sampai akhir dan pada akhirnya ketemu flag.txt

```
kali@kali: ~/LKS/Doll/_dolls.jpg.extracted/base_images/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted/base_images/_4_c.jpg.extracted
File Actions Edit View Help
3226      0xC9A      TIFF image data, big-endian, offset of first image directory: 8
123606    0x1E2D6    Zip archive data, at least v2.0 to extract, compressed size: 77650, uncompressed size
: 79807, name: base_images/4_c.jpg
201422    0x312CE    End of Zip archive, footer length: 22

(kali@kali)-[~/_/_dolls.jpg.extracted/base_images/_2_c.jpg.extracted/base_images]
$ ls
3_c.jpg _3_c.jpg.extracted

(kali@kali)-[~/_/_dolls.jpg.extracted/base_images/_2_c.jpg.extracted/base_images]
$ cd _3_c.jpg.extracted

(kali@kali)-[~/_/_base_images/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted]
$ ls
0 1E2D6.zip 312CE base_images C9A

(kali@kali)-[~/_/_base_images/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted]
$ cd base_images

(kali@kali)-[~/_/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted/base_images]
$ ls
4_c.jpg

(kali@kali)-[~/_/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted/base_images]
$ binwalk --extract --dd=".*" 4_c.jpg

DECIMAL      HEXADEDECIMAL  DESCRIPTION
-----
0            0x0            PNG image, 320 x 768, 8-bit/color RGBA, non-interlaced
3226         0xC9A         TIFF image data, big-endian, offset of first image directory: 8
79578        0x136DA       Zip archive data, at least v2.0 to extract, compressed size: 63, uncompressed size: 8
1, name: flag.txt
79785        0x137A9       End of Zip archive, footer length: 22

(kali@kali)-[~/_/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted/base_images]
$ ls
4_c.jpg _4_c.jpg.extracted

(kali@kali)-[~/_/_2_c.jpg.extracted/base_images/_3_c.jpg.extracted/base_images]
$ cd _4_c.jpg.extracted

(kali@kali)-[~/_/_base_images/_3_c.jpg.extracted/base_images/_4_c.jpg.extracted]
$ ls
0 136DA.zip 137A9 C9A flag.txt

(kali@kali)-[~/_/_base_images/_3_c.jpg.extracted/base_images/_4_c.jpg.extracted]
$ cat flag.txt
picoCTF{96fac089316e094d41ea046900197662}

(kali@kali)-[~/_/_base_images/_3_c.jpg.extracted/base_images/_4_c.jpg.extracted]
```

picoCTF{96fac089316e094d41ea046900197662}

## Magikarp Ground Mission

*Do you know how to move between directories and read files in the shell? Start the container, `ssh` to it, and then `ls` once connected to begin. Login via `ssh` as `ctf-player` with the password, `481e7b14`*

disini langsung saya connect ke ssh sesuai perintah. Didalam ssh disediakan berbagai txt sebagai instruksi dimana flag dan berada dan yang perlu dilakukan hanyalah cat dan juga change directory sesuai instruksi lalu menggabungkan part part flag

```
ctf-player@pico-chall$ ls
10f3.flag.txt  instructions-to-20f3.txt
ctf-player@pico-chall$ cat 10f3.flag.txt
picoCTF{xxsh_
ctf-player@pico-chall$ cat instructions-to-20f3.txt
Next, go to the root of all things, more succinctly `/'
ctf-player@pico-chall$ cd /
ctf-player@pico-chall$ ls
20f3.flag.txt  bin  boot  dev  etc  home  instructions-to-30f3.txt  lib  lib64  media  mnt  opt  proc  root  run  sb
ctf-player@pico-chall$ cd 20f3.flag.txt
-bash: cd: 20f3.flag.txt: Not a directory
ctf-player@pico-chall$ cat 20f3.flag.txt
0ut_of_\V/4t3r_
ctf-player@pico-chall$ cat instructions-to-30f3.txt
Lastly, ctf-player, go home ... more succinctly `~`
ctf-player@pico-chall$ cd ~
ctf-player@pico-chall$ ls
30f3.flag.txt  drop-in
ctf-player@pico-chall$ cat 30f3.flag.txt
1118a9a4}
ctf-player@pico-chall$
```

didapatkan lah:

picoCTF{xxsh\_0ut\_of\_\V/4t3r\_1118a9a4}