

Nexa x MMU CTF 2022 Part 1

Author : yialexlee / w9u0l1.l2lvi

Reverse Engineering

Crack Me



Download the python file and open it. We found the encrypted flag

```
# We want our biggest client to know his information is safe with us.  
bezos_cc_secret = "E<08R)&+@6:8E8@T"  
  
# Reference alphabet
```

After look around the source code, we decode the encrypted flag becos_cc_secret.

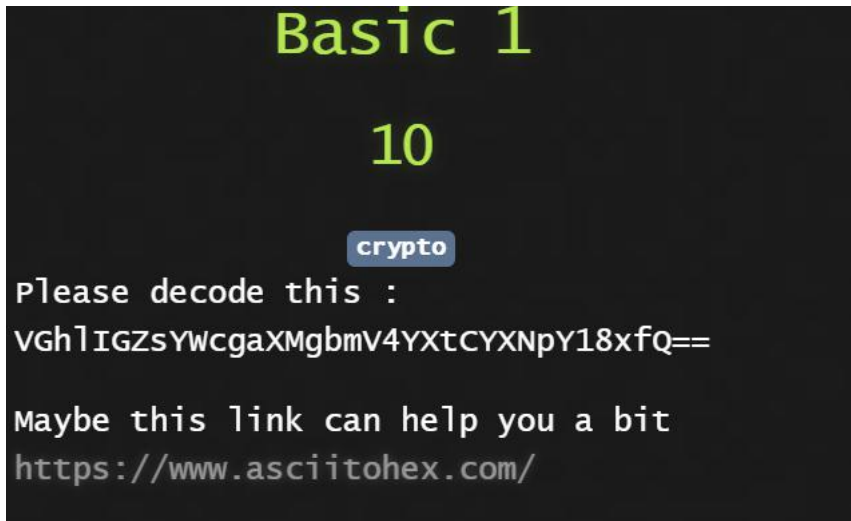
```
    + str(greatest_value) )  
  
decode_secret(becos_cc_secret)
```

After run, we get the flag.

```
Install the latest PowerShell for new features and improvements  
PS C:\Users\lee52\Downloads> & 'C:\Users\lee52\AppData\Local\Microsoft\Windows\CurrentVersion\Installer\x-wwz\PowerShell\PowerShell.exe' -ExecutionPolicy AllSigned -File 'c:\Users\lee52\Downloads\aa84f29754b12220ea353db9c2867fa  
nixa{ROTi_canai}  
PS C:\Users\lee52\Downloads>
```

Cryptography

Basic 1



Its base64 encrypted, so just use any base64 decoder to decode

Decode from Base64 format

Simply enter your data then push the decode button.

VGhlIGZsYWcgaXMgbmV4YXtCYXNpY18xfQ==

DECODE

The flag is nexa{Basic_1}

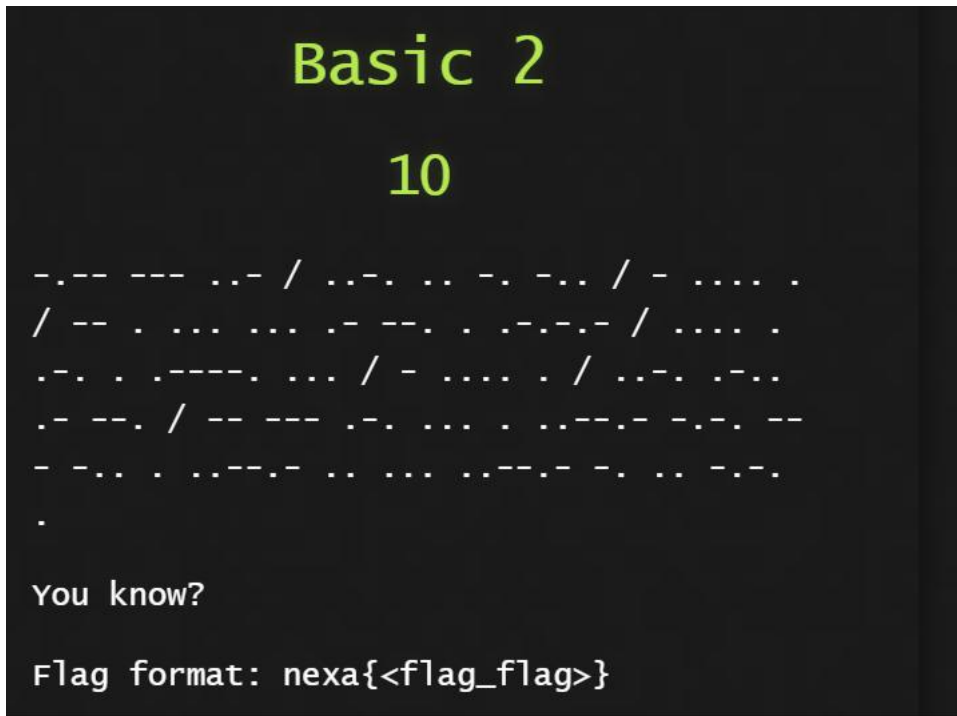
For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

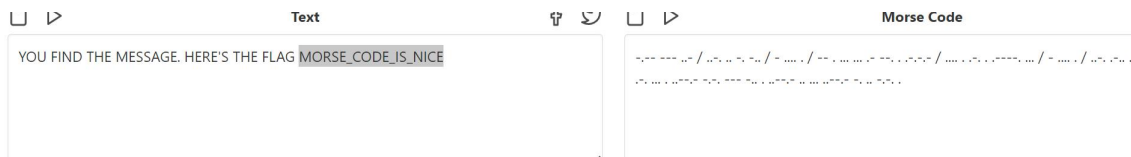
☐ Decode each line separately (useful for when you have multiple entries).

☒ Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

Basic 2



It is a morse code, so we use morse code decoder to decode



Nexa{MORSE_CODE_IS_NICE}

Intermediate 1

Intermediate 1

20

"Dear Decision maker ; Especially for you - this cutting-edge information . This is a one time mailing there is no need to request removal if you won't want any more . This mail is being sent in compliance with senate bill 1626 ; Title 6 , Section 304 . This is a legitimate business proposal . Why work for somebody else when you can become rich within 86 WEEKS . Have you ever noticed the baby boomers are more demanding than their parents plus more people than ever are surfing the web . I

We use spamimic to decode. But remember to remove the “ ”



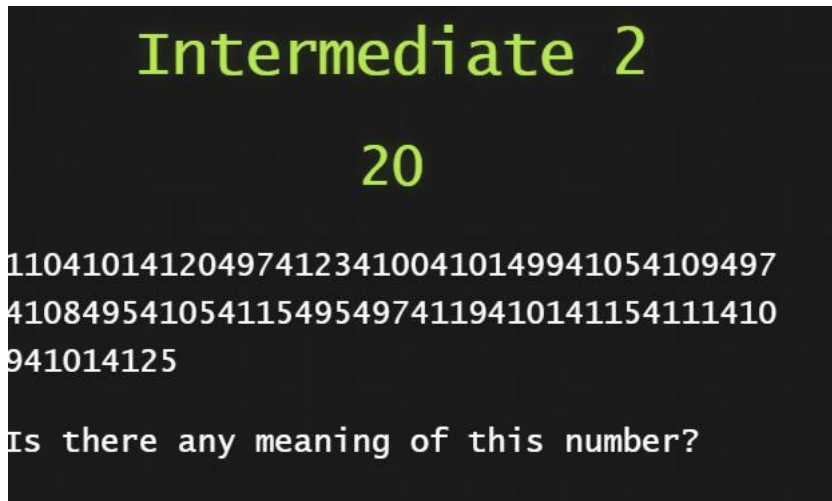
Decoded

Your spam message **Dear Decision maker ; Especially for you...**

Look wrong?, try the [old version](#)

Copyright © 2000-2020 spammimic.com, All rights reserved

Intermediate 2



Remove the number 4 replace with space. Now it's a decimal, we convert decimal to text.

INTERPRET AS
DECIMAL ▼

CONVERT TO
TEXT ▼

Separator

110 101 120 97 123 100 101 99 105 109 97 108 95
105 115 95 97 119 101 115 111 109 101 125

Transform

None ▼

nexa{decimal_is_awesome}

Intermediate 3

Intermediate 3

20

YXJrbntVbmVxX2xyZ19wbmFfb3JfZmJ5aXJxfQ==

Base64 is not enough to know the
message...help me!!

We use convert tool to decode, and the flag is shown at ROT13

61 72 6b 6e 7b 55 6e 65 71 5f 6c 72 67 5f 70 6e 61 5f
6f 72 5f 66 62 79 69 72 71 7d

Convert

Highlight Text

YXJrbntVbmVxX2xyZ19wbmFfb3JfZmJ5aXJxfQ==

Convert

Highlight Text

ROT13

nexa{Hard yet can be solved}

Cracking

Basic Wifi Cracking

Basic Wifi Cracking

50

Can you find the wifi password for me?
Flag Format: nexa{wifi_password}

 34cf7cd7...

Download the cap file and use aircrack-ng or hashcat to crack it. Here we use aircrack with wordlist (rockyou.txt) to crack the cap file

```
(root@kali)-[~]
# aircrack-ng /home/kali/Downloads/34cf7cd73063b5d98eed086b539dca96.cap -w /usr/share/wordlists/rockyou.txt
Reading packets, please wait...
Opening /home/kali/Downloads/34cf7cd73063b5d98eed086b539dca96.cap
Read 1033 packets.

# BSSID      ESSID      Encryption
1 6A:3E:26:EF:76:53 nexasecurenetworkkkk WPA (1 handshake)

Choosing first network as target.

Reading packets, please wait...
Opening /home/kali/Downloads/34cf7cd73063b5d98eed086b539dca96.cap
Read 1033 packets.

1 potential targets
```

```
Aircrack-ng 1.6

[00:31:44] 9770559/14344392 keys tested (5052.16 k/s)

Time left: 15 minutes, 5 seconds      68.11%

KEY FOUND! [ betoeresmivida12 ]

Master Key      : D7 DD EC 07 8D 08 3E 0A CF A8 9F D2 1B 55 A5 EC
                  1F 94 94 55 5C 5A 10 7F B9 C8 90 67 79 B5 A2 9E

Transient Key    : 70 86 5D 38 89 9E E9 12 93 2A 08 99 C5 BD 97 48
                  80 2F 64 21 1C 81 01 1D BE 69 85 AF B7 9F 25 E6
                  D9 49 04 C8 35 07 54 D5 16 27 88 87 52 E9 3F E2
                  96 95 9F 2C 38 C5 A2 49 77 06 9A 89 01 2D 89 00

EAPOL HMAC      : B7 82 C5 21 6F A8 1C 76 D4 B1 96 BD 2A CD 24 A3
```

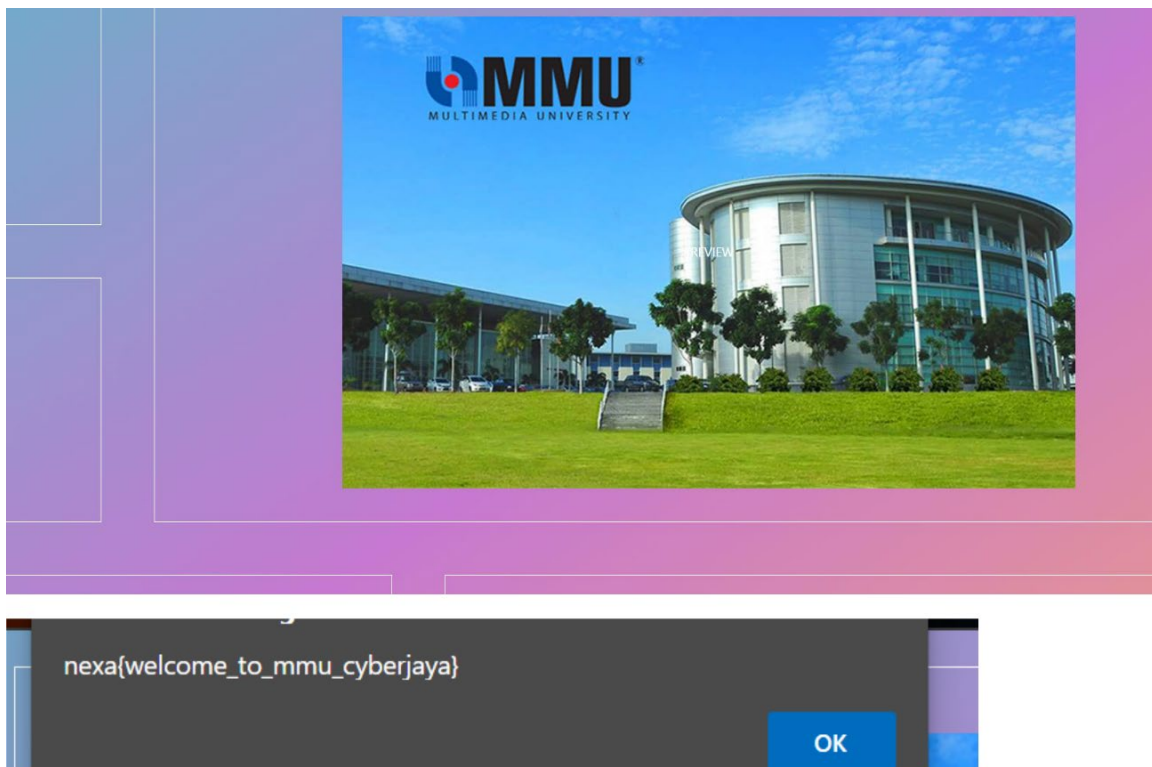
And we get the password betoeresmivida12 so the flag is nexa{betoeresmivida12}

Steganography

Basic 5



Download the image file and use hide text in image extension to solve it



Intermediate 4



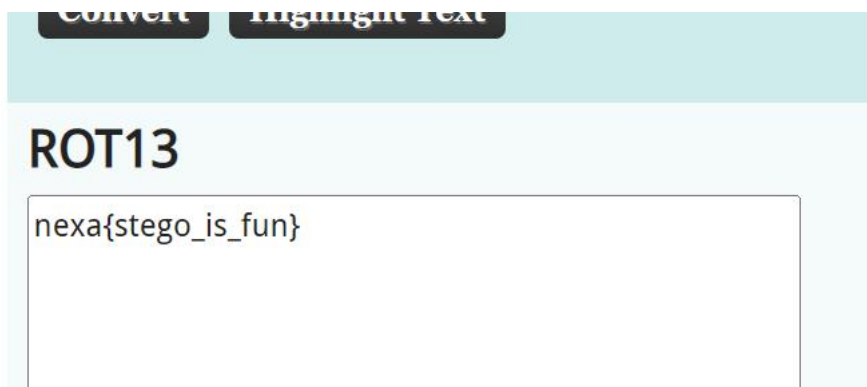
Download the image and use steghide to extract the flag.txt file inside. Based on the hint given, we guess the pass is admin, and it really is.

```
(root@kali)~[~]
# steghide extract -sf /home/kali/Downloads/RookChessmon.jpeg
Enter passphrase:
wrote extracted data to "flag.txt".

(root@kali)~[~]
# cat flag.txt
arkn{fgrtb_vf_sha}

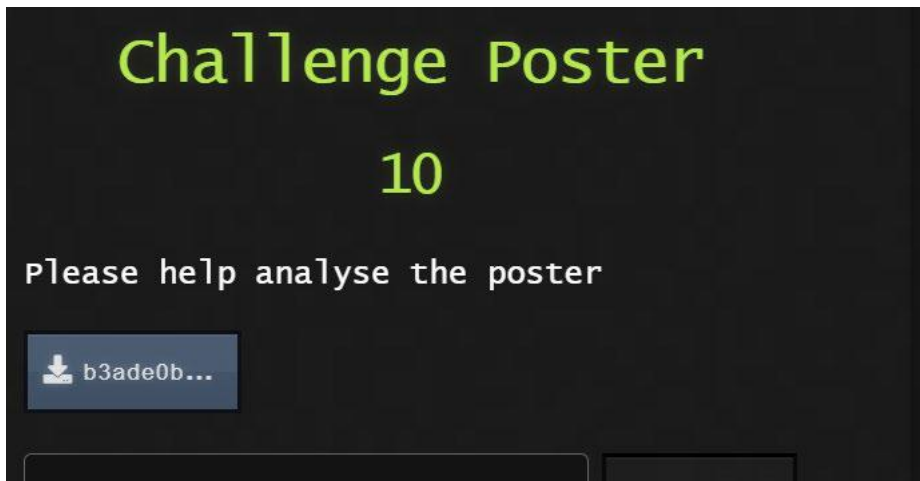
(root@kali)~[~]
```

We get the fake flag. After some trying we found that this fake flag is encrypted version of real flag. We get the real flag after ROT13 decode

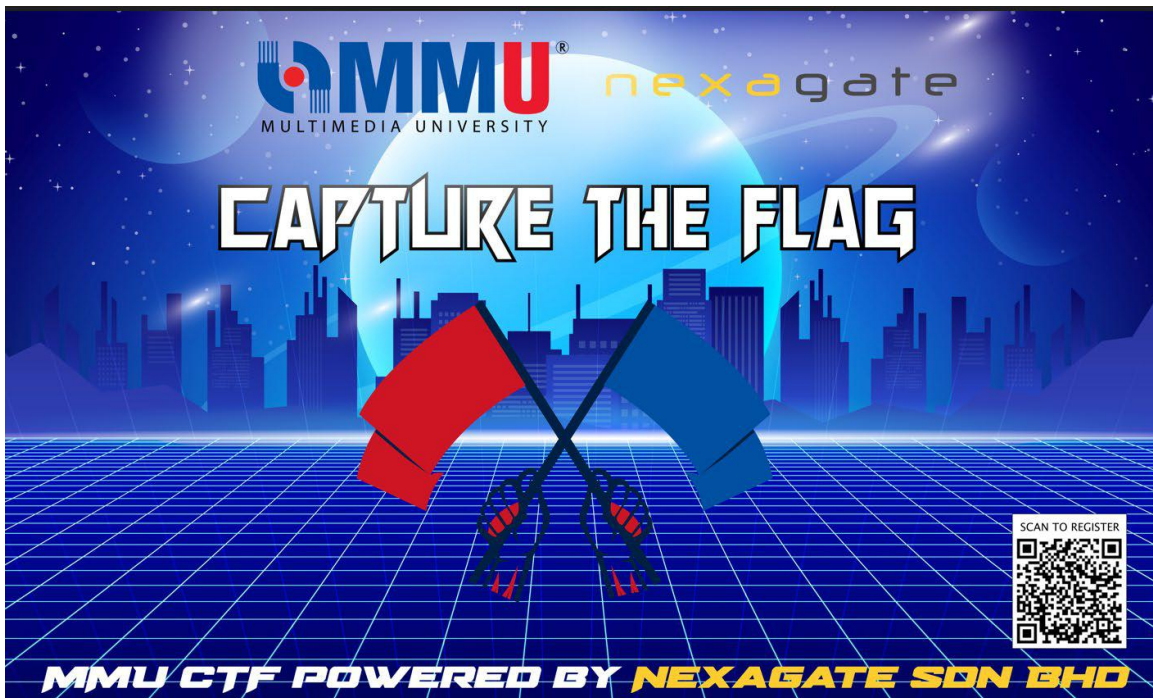


MISC

Challenge Poster

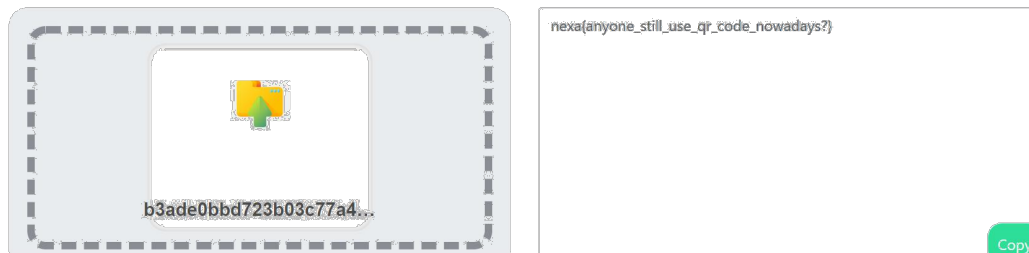


Download the poster

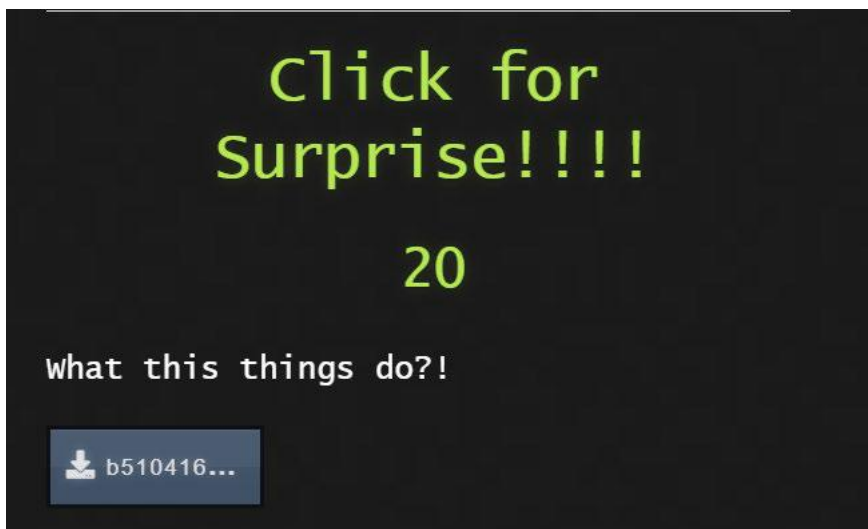


Scan the qrcode and get the flag

QR Scanner with file just drag and drop or select your file



Click for surprise



I really run the exe file at first..... Its shutdown my laptop. So I sure this is a bat that contains shutdown command. So this times I use notepad++ to open, and the flag is there.

```
@echo off
::nexa(editnotopen)
shutdown -s -f -t 5 -c "You're in Cybersecurity and you execute random Batch scripts?"
```

h3x8d1mdkw8fnc1



I forgot to screenshot, but I installed the application at first. After install there is a picture on my desktop, and nothing. So I looking for the directory of the application, there is a flag.html but it's a trap haha. After checking all the file in application directory and found nothing, I uninstall the application and reinstall again to make sure I did not miss something. And we found the flag in the process of installation, at the T&C part. Ok, interesting. Next time I will read the T&C haha.

License Agreement

Read the following important information before continuing.



Please read the following License Agreement. you must accept the terms of this agreement before continuing with the installation.

Flag for CTF

The Company prepared flag for you which is nexa {have_you_ever_read_terms_and_conditions?}.

United States Legal Compliance

You represent and warrant that (i) You are not located in a country that is subject to the United States government embargo, or that has been designated by the United States government as a "terrorist supporting" country, and (ii) You are not listed on any

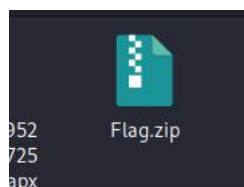
Unknown file type



Use winhex to check the hex, we see PK and flag.txt inside. So we know this is a zip file and its contain a flag.txt inside. But the magic header is empty. So, we add the missing PK at the beginning.

1E 1F	ANSI ASCII
46 6C	PK - zTRC[& Fl
3B 94	ag.txt)2Áûá +-^4~R Wé Últ3\$;"
00 00	m' é³PK ? - zTRC[&
00 00	\$ Flag.txt
50 4B	ÿç-9q@ø ¹éD`q@ø ò-Jöp@ø PK
	Z D

Now, the hex is correct now. So we change the file extension to .zip



We try to extract but it required password. So, we use john the ripper to crack it. The password is 1234

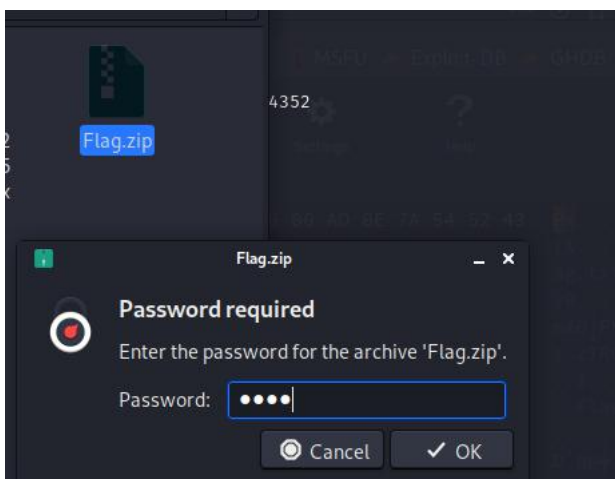
```

(root@kali)-[~]
# zip2john /home/kali/Downloads/Flag.zip > flag.txt
ver 2.0 Flag.zip/Flag.txt PKZIP Encr: cmplen=30, decmplen=18, crc=265B4352

(root@kali)-[~]
# john flag.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
1234 (Flag.zip/Flag.txt)
1g 0:00:00:00 DONE 2/3 (2022-03-26 10:19) 33.33g/s 1594Kp/s 1594Kc/s 1594KC/s 123456..ferrises
Use the "--show" option to display all of the cracked passwords reliably
Session completed

```

Extract the zip file with the password we cracked:1234

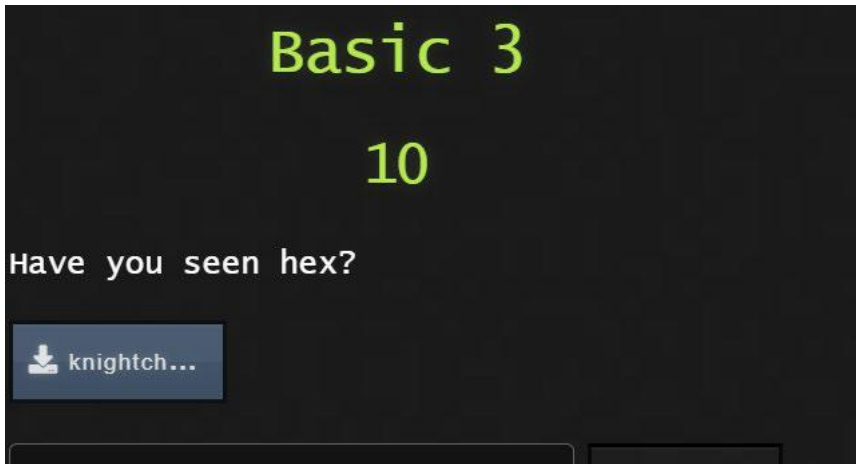


Open the flag.txt and we get the flag

```
nexa{magic_header}
```

Forensic

Basic3



Use winhex to check the hex and we can see the flag is already there

```
00  ÿøÿà JFIF      d d  ÿì Ducky
01      d  ÿì Adobe dÀ  ÿŮ "
01  nexa{stego_basic}
01
03
```

Basic4



Same, we use winhex to check the hex first.

1B	1C	1D	1E	1F	ANSI ASCII
01	02	01	01	02	%PNG JFIF ' ' yÜ c
08	09	0B	09	08	
00	43	01	02	02	yÜ c
0C	0C	0C	0C	0C	
0C	0C	0C	FF	C0	yÄ
01	01	01	01	00	~ e " yÄ
03	02	04	03	05	yÄ µ
81	91	A1	08	23	} !1A Qa "q 2 'i #
36	37	38	39	3A	B±Ä RÑ\$3br, &'()*456789:
76	77	78	79	7A	CDEFGHIJSTUVWXYZcdefghijstuvwxxyz
B3	B4	B5	B6	B7	f.....t+~%\$'"".....~%\$&~Y!\$'C~..µ!
E7	E8	E9	EA	F1	..°AAAAAÇEEEOOOOO°×UUUaaaaaaçeeen
00	00	00	00	01	ööööö÷öüüYÄ
00	01	02	77	00	yÄ µ w
23	33	52	F0	15	!1 AÖ ad "2 B'±Ä #3Rö

We can see the magic header is png , but there is a JFIF . So we guess that is a hint to tell us this file is not png, it's a jfif. So, we change the magic header to jfif magic header and save as jpg

[illegible]

Open the jpg and we get the flag

nexa{magic_number}

Normal PCAP: Part 1



The hint mention it is a webserver, so we search http and https, and we found flag in one of the http packet

```
query 0xa568 A incoming.telemetry.mozilla.org OPT
query 0x3ab3 AAAA incoming.telemetry.mozilla.org OPT
n Data
n Data
ACK1 Seq=1 Ack=152 Win=64240 Len=0
```

on the page, just like this `nexa{s1mpL3_w38_w1tH_l0v3}` tag and its



Normal PCAP: Part 2

Normal PCAP: Part 2

20

Find the flag on the web attachment!
(use same attachment from "Normal PCAP:
Part 1")

The hint mention it's a web attachment, so we find all the web attachment in this pcap file

Packet	Hostname	Content Type	Size	Filename
26	192.168.137.130	text/html	311 bytes	\
51	192.168.137.130	text/html	277 bytes	favicon.ico
3501	192.168.137.130	image/png	10 MB	download.png

After looking for favicon.ico and download.png, we found flag in download.png



Shift your focus

Shift your focus

20

Find the *flag* in this text document!!!!

 43cddc4...

Use winhex to left shift by 1 bit

C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	AN
6	31	B7	B6	B2	90	3A	37	90	33	37	B9	32	B7	39	B4	B1	90	31	B4	\$2¶67 0·2 ;²¶1·¶² :7 37¹
9	90	34	B9	90	35	3A	B9	BA	10	33	34	B6	36	32	B9	10	3A	32	BC	0¶62·3²— *44¹ 4¹ 5:¹° 34
0	36	37	B7	33	B2	B9	17	06	85	06	85	37	32	BC	30	BD	B0	AF	B1	: :7 6°µ² 4° 67·3²¹
																				4°/°94±µ¼¾

Modify Data

☐ Add

0

☐ hexadecimal

Integer type:

...

8 bit, signed

Value range:

☐ stay within limits

☒ allow over/underflow

☐ Reverse byte order

4 bytes

☐ Invert bits

☒ Left shift by 1 bit

☐ XOR 00

☐ Right shift by 1 bit

☐ OR 00

☐ Shift by -1 bytes

☐ AND 00

☐ Circular left rotation

☐ ROT13

OK

Cancel

Help

And we can see the flag

```
hello and welcome to forensic ch  
allenge. This is just filler tex  
t to make it longer. nexa{a_b  
it_tricky|
```

Normal PCAP: Part 3

Normal PCAP: Part 3

30

Find the flag on the netcat
communication! (use same attachment from
"Normal PCAP: Part 1")

Hint mention the netcat, so we looking for icmp,tcp and udp. At the end we found
flag in one of the tcp packet

```
..). . . . . % . . . . .  
. SURE. IT nexa{  
nc_i5_@_ p0w3rfuL  
L_t00l}.
```

Normal PCAP: Part 4

Normal PCAP: Part 4

30

Find the flag on the telnet connection!
(use same attachment from "Normal PCAP:
Part 1")

Hint mentions the telnet, so we looking for telnet. And we found flag in one of the telnet packets

```

c0 a8  ..!)@.@. ....
80 18  .....R.S ...."....
2d 1d  ..P=..... ..%.`&-..
77 33  ..nexa{t 3ln3t_w3
33 7d  r3_n3v3r _s3cuR3}
75 40  ...]0;ne xabuntu@
3b 33  ubuntu: ~...[01;3
75 6e  2mnexabu ntu@ubun
34 6d  tu.[00m: .[01:34m
```