Steganography

...

What is Steganography?

What is Steganography?

Cryptography

Steganography

What is Steganography?

Cryptography

Let’s use strong keys and algorithms, so no one can break it!

Steganography

What is Steganography?

Cryptography

Let’s use strong keys and algorithms, so no one can break it!

Steganography

Let’s hide our data, so no one will ever try to break anything!

What is Steganography?

Cryptography

Let’s use strong keys and algorithms, so no one can break it!

Steganography

Let’s hide our data, so no one will ever try to break anything!

Steganography is good, when it’s hard for attacker to tell if there is data hidden

Some examples

Data inside data

Least significant bit steganography

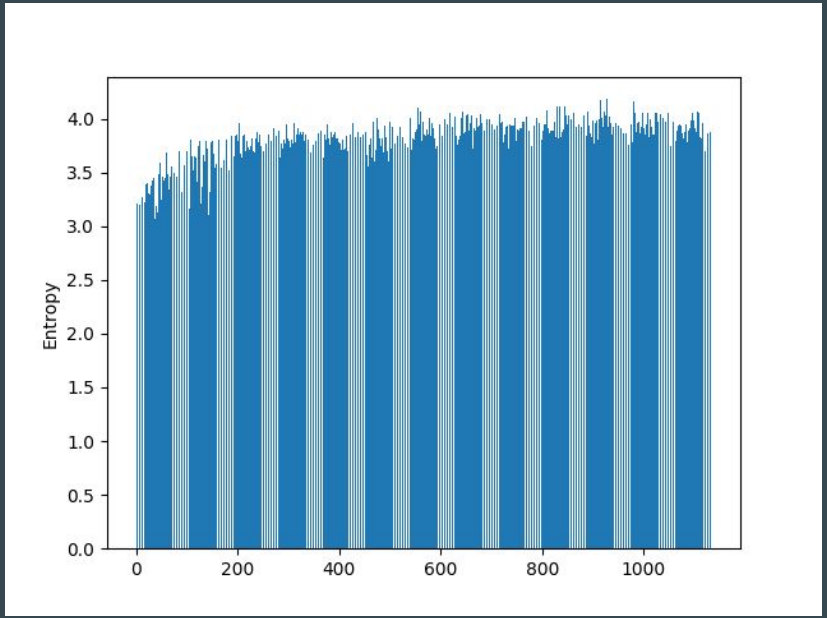
Least significant bit steganography

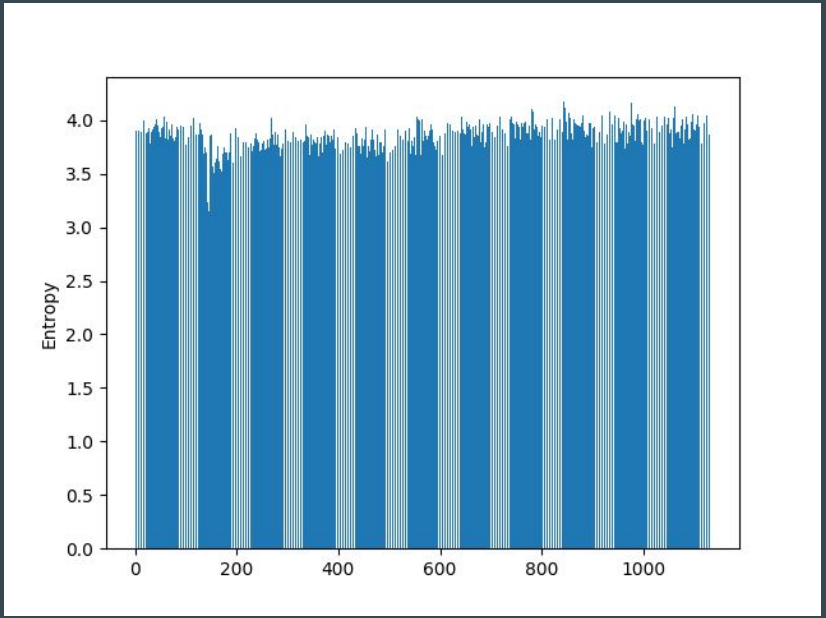
#E80808 - 

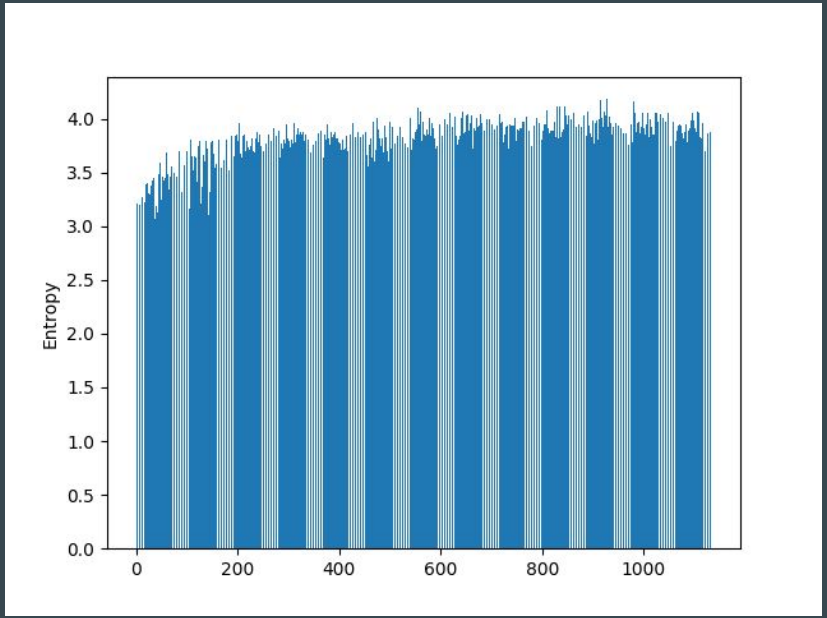
#E80809 - 

Original

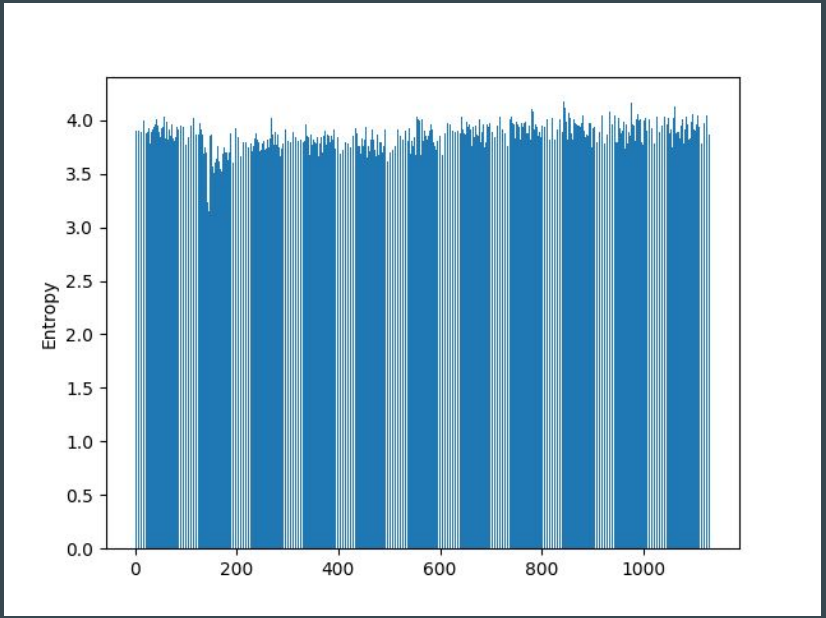
After encoding


Entropy
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
0
200
400
600
800
1000


Entropy
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
0
200
400
600
800
1000



No encryption



Encryption

Data inside metadata

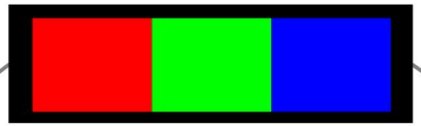
PNG file format

PNG file format

Before each scanline, there is one byte, indicating which filter algorithm was used

PORTABLE NETWORK GRAPHICS

ANGE ALBERTINI
http://www.corkami.com



```
    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00: 89 .P .N .G 0D 0A 1A 0A 00 00 00 0D .I .H .D .R
10: 00 00 00 03 00 00 00 01 08 02 00 00 00 94 82 83
20: E3 00 00 00 15 .I .D .A .T 08 1D 01 0A 00 F5 FF
30: 00 FF 00 00 00 FF 00 00 00 FF 0E FB 02 FE E9 32
40: 61 E5 00 00 00 00 .I .E .N .D AE 42 60 82
```

SIGNATURE

FIELDS	VALUES
signature	\x89 PNG \r\n \x1a \n

HEADER

size	0x0000000D
id	IHDR
width	0x00000003
height	0x00000001
bpp	0x08
color	0x02 RGB
compression	0x00 DEFLATE
filter	0x00
interlace	0x00
CRC32	0x948283E3

DATA

	size	0x00000015
	id	IDAT
ZLIB	window size	0b0000**1000**
ZLIB	method	0b0000**1000** DEFLATE
ZLIB	level / dict.	0b00011101
ZLIB	checksum	0x081D % 31 = 0
DEFLATE	last block	0b0000000**1** FINAL
DEFLATE	block type	0b000000**01** RAW
DEFLATE	data length	0x000A
DEFLATE	!length	0xFFF5
PIXELS	line filter	0x00 NONE
PIXELS	FF 00 00 00 FF 00 00 00 FF	
	adler32	0x0EFB02FE
	CRC32	0xE93261E5

END

size	0x00000000
id	IEND
CRC32	0xAE426082

PNG file format

Before each scanline, there is one byte, indicating which filter algorithm was used

Each algorithm is reversible

PNG file format

Before each scanline, there is one byte, indicating which filter algorithm was used

Each algorithm is reversible

Let’s hide some data there!

5 available filter algorithms

5 available filter algorithms

~$\log_2 5$ * height bits

5 available filter algorithms

~$\log_2 5$ * height bits

For example, for 1920x1080 image, we get ~313 bytes

Lossy compression?

Lossy compression?

No Problem!

JPEG compression algorithm

JPEG compression algorithm

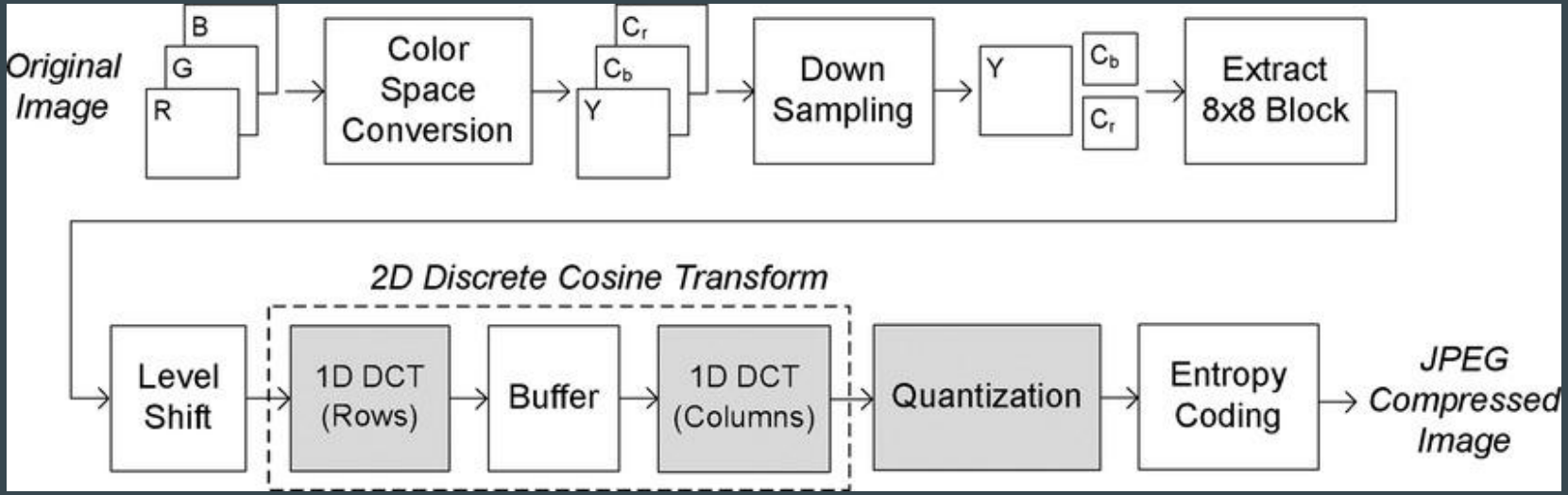


image source: https://www.researchgate.net/figure/JPEG-compression-algorithm_fig1_224440999

JPEG compression algorithm

After DCT

-376	-23	1	-2.5	-0.3	4	0.2	-2.6
-224	53	20	3.4	5	3	0.6	2.3
68	3.3	-14	-0.3	-2.8	-1.9	-4.7	-6.2
2.3	-8.9	-1.5	-3.8	-2.5	1.2	1.4	1.9
-8.4	1.2	1.9	3.3	-2.1	5	1.8	5.3
4.5	7.3	-7.4	1.9	1.3	-0.7	-1.5	-6
6.4	6.8	-3.2	-2.6	1.3	-2.1	1.7	1
-16	0.1	9	0.8	1.8	1.7	-1	1

Quantization Table

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

After Quantization

-24	-23	0	0	0	0	0	0
-19	4	1	0	0	0	0	0
5	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

image source: http://arjunsreedharan.org/post/146070390717/jpeg-101-how-does-jpeg-work

DCT LSB encoding

DCT LSB encoding

- Encode lsb of the message into lsb of each value in quantized DCT coefficient

DCT LSB encoding

- Encode lsb of the message into lsb of each value in quantized DCT coefficient
- Can’t use 1

DCT LSB encoding

- Encode lsb of the message into lsb of each value in quantized DCT coefficient
- Can’t use 1
- Can’t use 0

Outguess

Outguess

Apart from hiding data in LSB of quantized DCT coefficients, it also changes some values, to preserve the histogram, making statistical attacks harder.

More info

Advanced JPEG Steganography and Detection by John Ortiz (https://www.youtube.com/watch?v=BQPkRlbVFEs)

Use cases?

Use cases?

- Malware communication

Blackhat 2016 conference

Blackhat 2016 conference

Two malware researchers(Pierre-Marc-Bureau & Christian Dietrich), shown there is malware, that uses steganography to communicate.

Gozi(Neverquest)

Gozi(Neverquest)

- Uses HTTPS as main communication channel

Gozi(Neverquest)

- Uses HTTPS as main communication channel
- Downloads favicon.ico via TOR

Gozi(Neverquest)

- Uses HTTPS as main communication channel
- Downloads favicon.ico via TOR
- Decodes real message, using LSB steganography

Gozi(Neverquest)

- Uses HTTPS as main communication channel
- Downloads favicon.ico via TOR
- Decodes real message, using LSB steganography
- Decrypts message using RC4 algorithm

Use cases?

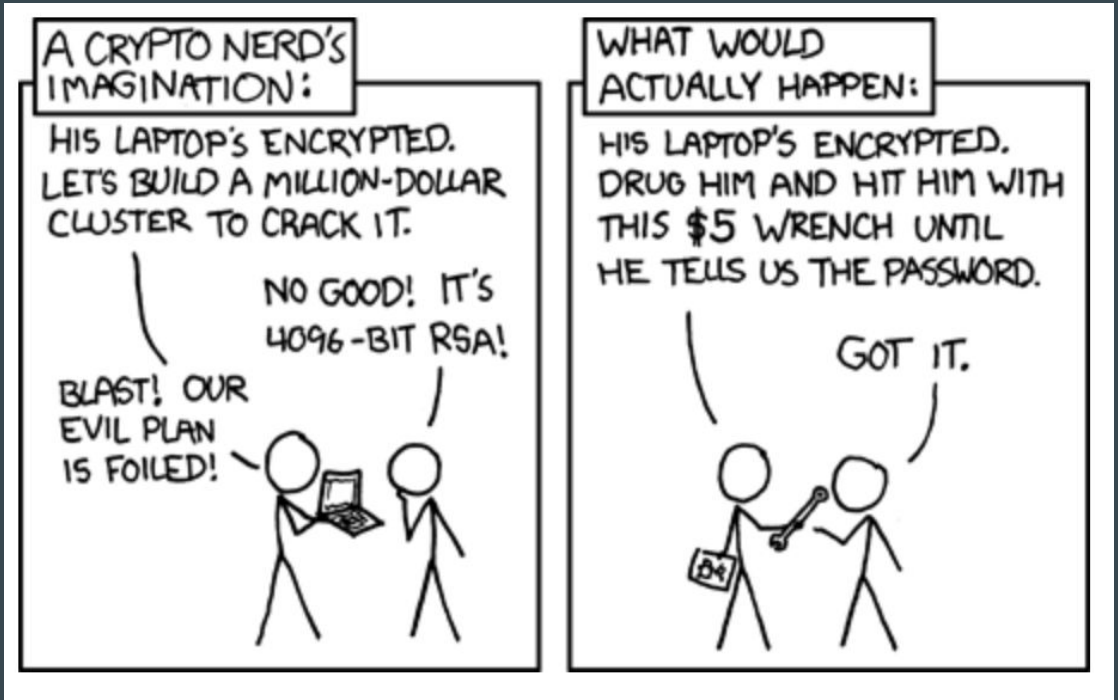
- Malware communication

Use cases?

- Malware communication
- Hidden volumes

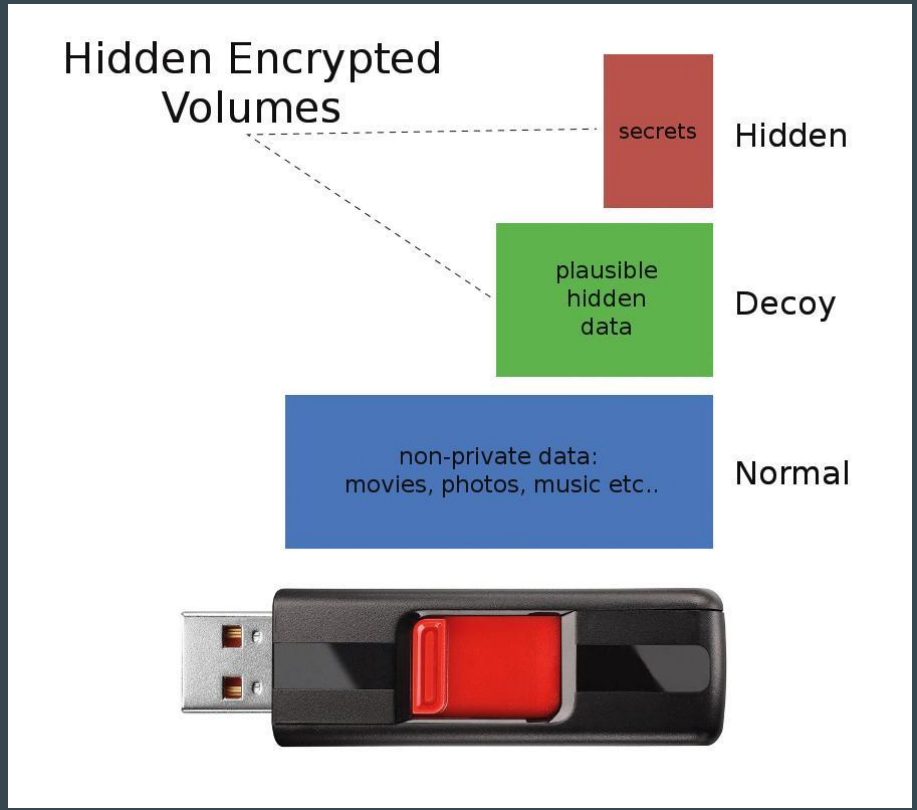
Rubber-house cryptanalysis

Rubber-house cryptanalysis



Solution?

Solution?


Hidden Encrypted Volumes
secrets
Hidden
plausible hidden data
Decoy
non-private data: movies, photos, music etc..
Normal

Tools

Tools

Windows - Veracrypt

Tools

Windows - Veracrypt

Linux - You have to do it by hand

Not as good as it looks like...

Not as good as it looks like...

- Cannot write too much to outer volume

Not as good as it looks like...

- Cannot write too much to outer volume
- Access date in outer volume inodes might seem suspicious

Not as good as it looks like...

- Cannot write too much to outer volume
- Access date in outer volume inodes might seem suspicious
- Need to watch out for bash history, and possibly some other logs

Source

https://www.linuxvoice.com/hidden-encrypted-volumes-keep-data-safe-and-secret/

Use cases?

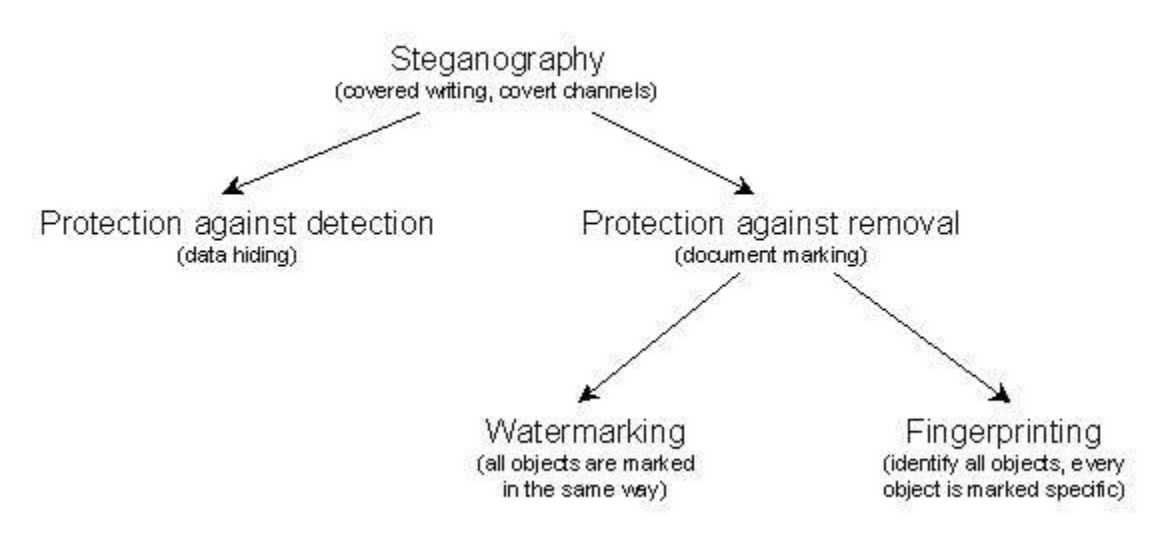
- Malware communication
- Hidden volumes

Use cases?

- Malware communication
- Hidden volumes
- Watermarking

Watermarking

Watermarking



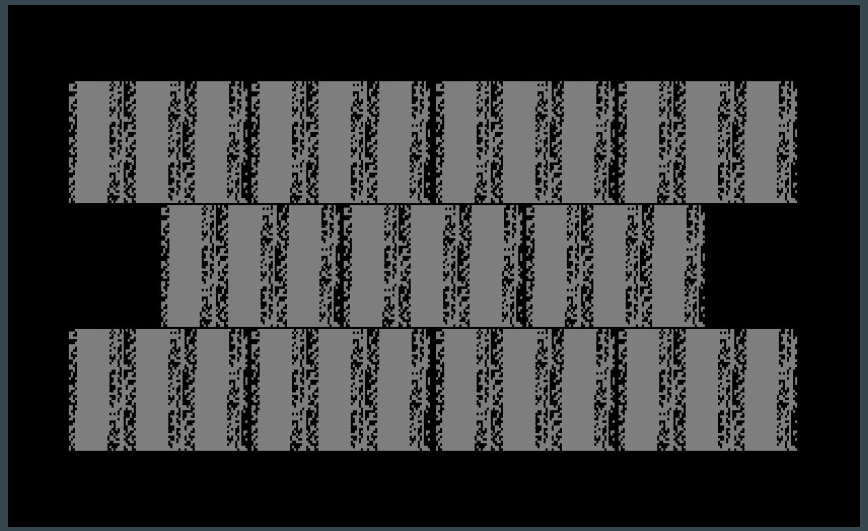
World of Warcraft

World of Warcraft

In 2012, some player found out weird pattern on his screenshots

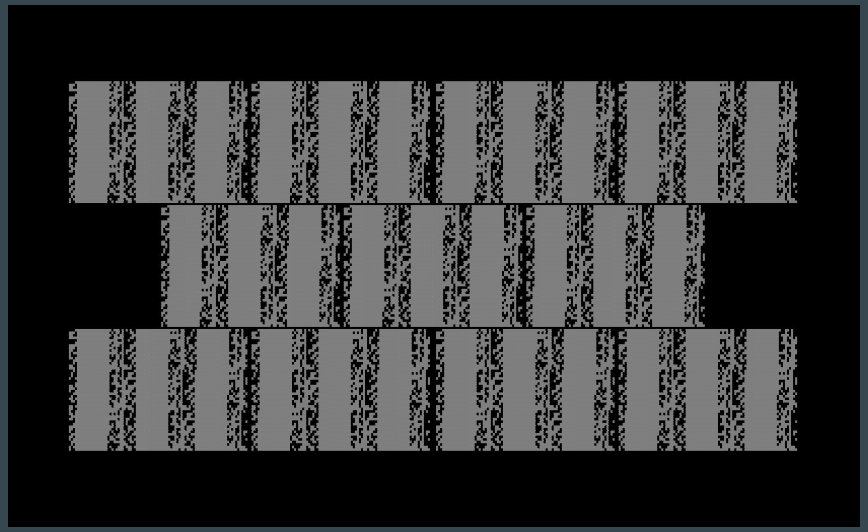
World of Warcraft

In 2012, some player found out weird pattern on his screenshots



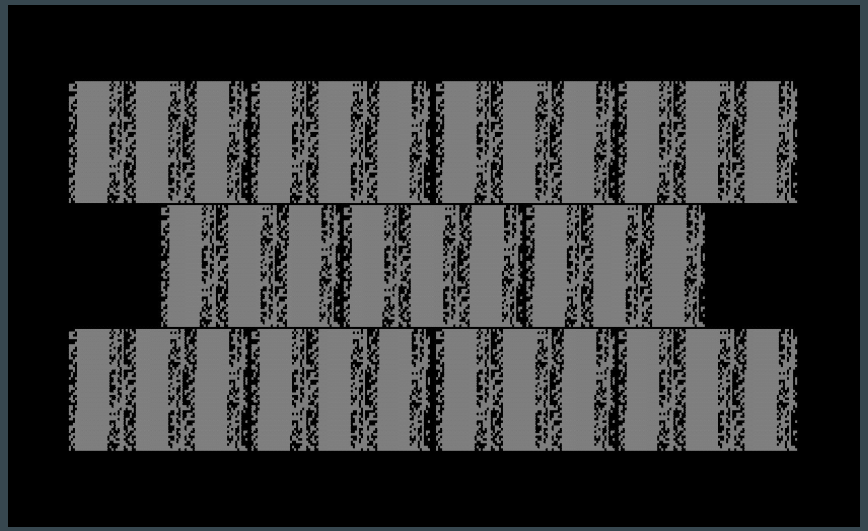
World of Warcraft

In 2012, some player found out weird pattern on his screenshots



World of Warcraft

In 2012, some player found out weird pattern on his screenshots



https://zaufanatrzeciastrona.pl/post/zrzuty-ekranu-world-of-warcraft-zawieraja-zakodowane-dane-uzytkownika/

World of Warcraft

Turns out, it contained user id, date, and server IP address

World of Warcraft

Turns out, it contained user id, date, and server IP address

Probably, Blizzard used this to identify people leaking information from closed testing phase

Easter Egg

Easter Egg

Blizzard also used similar technique, as an easter egg in Diablo

Easter Egg

Blizzard also used similar technique, as an easter egg in Diablo I

How to destroy (possibly) hidden data?

How to destroy (possibly) hidden data?

- For JPG - remove one line, so DCT coefficients would change

How to destroy (possibly) hidden data?

- For JPG - remove one line, so DCT coefficients would change
- For lossless LSB encoding, put random values to pixels LSB

Questions?