

## Crypto

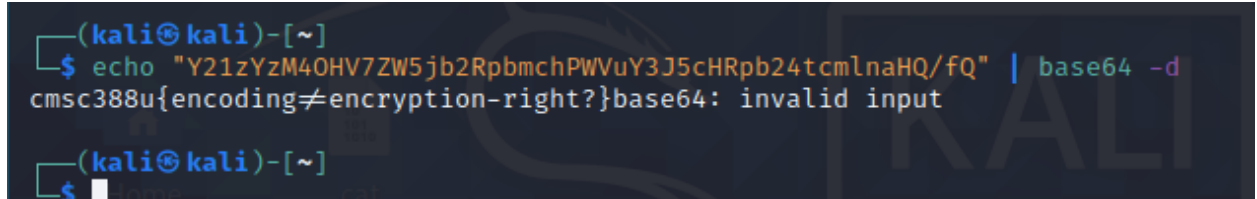
### 1. a. The text

Y21zYzM4OHV7ZW5jb2RpbmchPWVuY3J5cHRpb24tcmlnaHQ/fQ

Is in Base64 so using the command

```
echo "Y21zYzM4OHV7ZW5jb2RpbmchPWVuY3J5cHRpb24tcmlnaHQ/fQ" |  
base64 -d
```

We get

A terminal window with a dark background and light-colored text. The prompt is (kali㉿kali)-[~]. The user enters the command: echo "Y21zYzM4OHV7ZW5jb2RpbmchPWVuY3J5cHRpb24tcmlnaHQ/fQ" | base64 -d. The output is: cm5c388u{encoding!=encryption-right?}base64: invalid input. The prompt is shown again at the bottom.

```
(kali㉿kali)-[~]  
$ echo "Y21zYzM4OHV7ZW5jb2RpbmchPWVuY3J5cHRpb24tcmlnaHQ/fQ" | base64 -d  
cm5c388u{encoding!=encryption-right?}base64: invalid input  
  
(kali㉿kali)-[~]  
$
```

The plaintext is thus:

<REDACTED>{encoding!=encryption-right?}

### b. The text

Pzpf388h{prnfre\_jnfa'g\_gung\_fzneg}

Is in the format of <REDACTED>{text}. Since the numbers aren't changed, this must be a simple shift or replacement encoding.

I decided to try a caesar cipher first. Using <https://www.dcode.fr/caesar-cipher>

★ SEARCH A TOOL ON DCODE BY KEYWORDS:

e.g. type 'sudoku'

★ BROWSE THE [FULL DCODE TOOLS' LIST](#)

Results

Brute-Force mode: all shifts are tested, text is limited to the a few hundreds of characters. To find the full text back with punctuation and space, please indicate the correct shift found (+XX) in the form.

↑↓

+13 cmcs388u{ceaser\_wasn't\_that\_smart}

+17 yiyo388q{yawoan\_swoj'p\_pdwp\_oiwnp}

+12 dndt388v{dfbtfs\_xbto'u\_uibu\_tnbsu}

+24 rbrh388j{rtphgt\_lphc'i\_iwpi\_hbpgi}

+2 nxnd388f{npldpc\_hldy'e\_esle\_dxlce}

+19 wgwm388o{wyumyl\_qumh'n\_nbun\_mgulin}

+1 oyoe388g{oqmeqd\_imez'f\_ftmf\_eymdf}

+25 qaqq388i{qsogsf\_kogb'h\_hvoh\_gaofh}

+5 kuka388c{kmiamz\_eiav'b\_bpib\_auibz}

+11 eoeu388w{egcugt\_ycup'v\_vjcv\_uoctv}

+7 isiy388a{ikgykx\_cgyt'z\_zngz\_ysgxz}

+6 jtjz388b{jlhzly\_dhzu'a\_aoha\_zthya}

+21 ueuk388m{uyskwj\_oskf'l\_lzsl\_kesjl}

+18 xhxn388p{xzvnzm\_rvni'o\_ocvo\_nhvmo}

+9 gqgw388y{giewiv\_aewr'x\_xlex\_wqevx}

+20 vfv1388n{vxtlxk\_ptlg'm\_matm\_lftkm}

+14 blbr388t{bdzrdq\_vzrm's\_sgzs\_rlzqs}

Canva

Start designing for free

CAESAR CIPHER DECODER

★ CAESAR SHIFTED CIPHERTEXT

pzpf388h{prnfce\_infa'g\_qung\_fzneq}

KNOWING THE SHIFT: 3

TEST ALL POSSIBLE SHIFTS (BRUTE-FORCE ATTACK)

DECRYPT CAESAR CODE

See also: [ROT Cipher](#) – [Shift Cipher](#)

WITH A CUSTOM ALPHABET

★ ALPHABET ABCDEFGHIJKLMNOPQRSTUVWXYZ

★ USE THE ASCII TABLE AS ALPHABET ☐

DECRYPT

CAESAR ENCODER

★ CAESAR CODE PLAIN TEXT

647321

We get a shift of +13, and the plaintext is  
 <REDACTED>{ceaser\_wasn't\_that\_smart}

- Yes, the email [thanos@sna.pp](mailto:thanos@sna.pp) was compromised in the data breach. The hash associated with it is c010b29fb22c2d79ce9402f72a4987f5

First I decided to try and determine what type of hash it is. I ran hashid  
 'c010b29fb22c2d79ce9402f72a4987f5'

```
(kali㉿kali)-[~]
└─$ hashid 'c010b29fb22c2d79ce9402f72a4987f5'
Analyzing 'c010b29fb22c2d79ce9402f72a4987f5'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snefru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x
```

Next, I loaded the hash into a text file called 'hash.txt'. Next, I decided to use MD5 as the encoding since it is commonly used. Next, I ran the commands

```
sudo unshadow rockyou.txt hash.txt > passwd
john --show passwd
```

```
(kali㉿kali)-[~/Desktop]
└─$ sudo unshadow rockyou.txt hash.txt > passwd

(kali㉿kali)-[~/Desktop]
└─$ john --show passwd
bobe89!:NO PASSWORD
:NO PASSWORD::::
LLLL:NO PASSWORD;;
:NO PASSWORD:magick::
:NO PASSWORD::0000~~~
[!@#]Le@r[!@#]:NO PASSWORD
zincntido:NO PASSWORD
zAIYUMI:NO PASSWORD
```

```
182 password hashes cracked, 554 left
```

Then i typed

```
john --wordlist=rockyou.txt --format=Raw-MD5 hash.txt
```

```

(kali㉿kali)-[~/Desktop]
└─$ john --wordlist=rockyou.txt --format=Raw-MD5 hash.txt
Using default input encoding: UTF-8
Loaded 50 password hashes with no different salts (Raw-MD5 [MD5 32/32])
Remaining 28 password hashes with no different salts
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
ordnancel      (nas123@sna.pp)
adik91         (kopink5@sna.pp)
vkikpdkovt     (ocean24712@sna.pp)
pablomanuel    (chloelowie@sna.pp)
OSITOPANDA     (conniecute@sna.pp)
d3ath          (023173624@sna.pp)
Jamaal1        (darylkrystel@sna.pp)
y.w@k@786      (linger3@sna.pp)
tori027        (32868254@sna.pp)
pass2630259    (mydangiu@sna.pp)
obeliks        (07080520@sna.pp)
marcshecanapish (issac0812@sna.pp)

```

I then ran `john --show --format=Raw-MD5 hash.txt` to see all of the cracked passwords in a readable format.

```

(kali㉿kali)-[~/Desktop]
└─$ john --show --format=Raw-MD5 hash.txt
0843089856@sna.pp:3655848
09265481511@sna.pp:lysabug17
camilaniky@sna.pp:330084
60985bucaneros15@sna.pp:canoramos
iloveang*@sna.pp:0851902709
va40ni@sna.pp:muephy
nayeliscarlett@sna.pp:5261192
PORelmundo8@sna.pp:mindy_
thanos@sna.pp:decieann07
raremen@sna.pp:boing08
lanikque@sna.pp:lufalufa
NOT4USE@sna.pp:marshii05
culobonito@sna.pp:ianosalina1991
Falcons!@sna.pp:smfintotihakeri
CIPHERUS@sna.pp:0826365647
doobie4@sna.pp:flaboi99
supercalypso@sna.pp:032406133
0815383375@sna.pp:sofimcevov
sorryjose@sna.pp:ni1994

```

The plaintext password for [thanos@sna.pp](mailto:thanos@sna.pp) is decieann07