INPT

CIT

CRYPTOGRAPHY

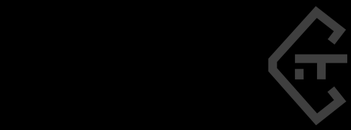
MADE BY:

FATIMA EZZAHRA ACHAIT

MOHAMMED KHALDOUNE

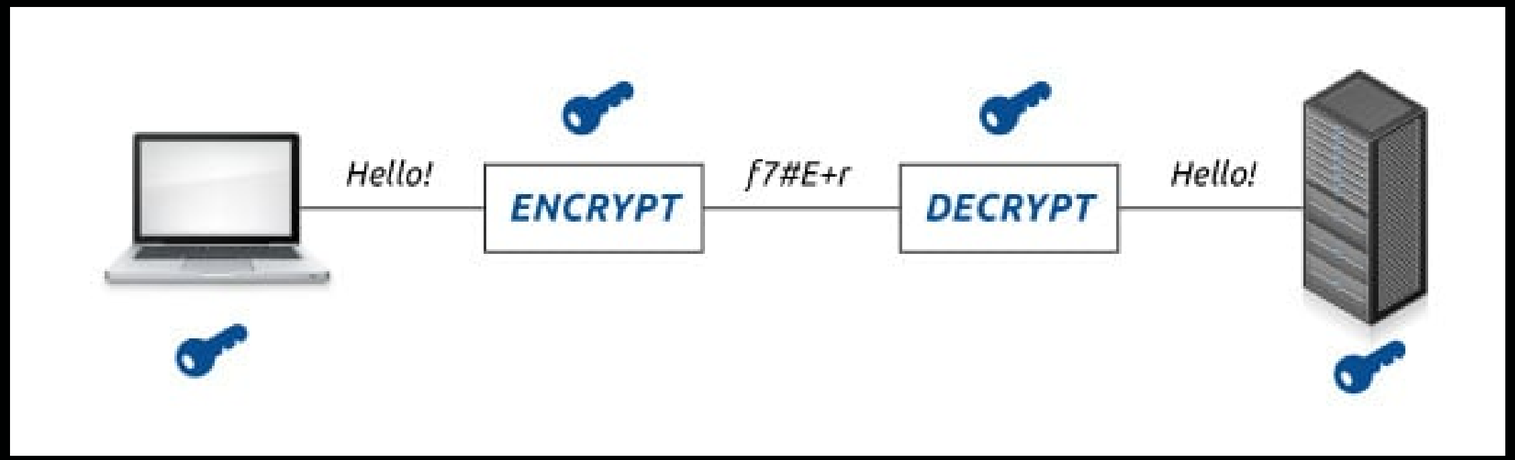
PLAN

1. WHAT IS CRYPTOGRAPY
2. KEY TERMS
2. CIA TRIAD
3. ENCODING VS ENCRYPTION VS HASHING
4. DECODING/ENCODING
    - 4.1 BASE32 & BASE64
    - 4.2 CAESAR CIPHER
    - 4.3 OTHERS
5. DECRYPTION/ENCRYPTION
    - 5.1 SYMMETRIC ENCRYPTION
    - 5.2 ASYMMETRIC ENCRYPTION : RSA
6. HASHING
    - 6.1 HASHCAT/JOHN THE RIPPER



WHAT IS CRYPTOGRAPHY?

Cryptography is transformation of data into another format in such a way that only specific individual(s) can reverse the transformation



KEY TERMS

- **Ciphertext** - The result of encrypting a plaintext, encrypted data
- **Cipher** - A method of encrypting or decrypting data. Modern ciphers are cryptographic, but there are many non cryptographic ciphers like Caesar.
- **Plaintext** - Data before encryption, often text but not always. Could be a photograph or other file
- **Key** - Some information that is needed to correctly decrypt the ciphertext and obtain the plaintext.
- **Passphrase** - Separate to the key, a passphrase is similar to a password and used to protect a key.
- **Brute force** - Attacking cryptography by trying every different password or every different key
- **Cryptanalysis** - Attacking cryptography by finding a weakness in the underlying maths

CIA TRIAD


1
CONFIDENTIALITY
INFORMATION SECURITY
2
INTEGRITY
3
AVAILABILITY

The CIA Triad

What Is the CIA?

Confidentiality	Integrity	Availability
I send you a message, and no one else knows what that message is.	I send you a message, and you receive exactly what I sent you	I send you a message, and you receive it

What's The Purpose of the CIA?

Data is not disclosed	Data is not tampered	Data is available

ENCODING VS ENCRYPTION VS HASHING

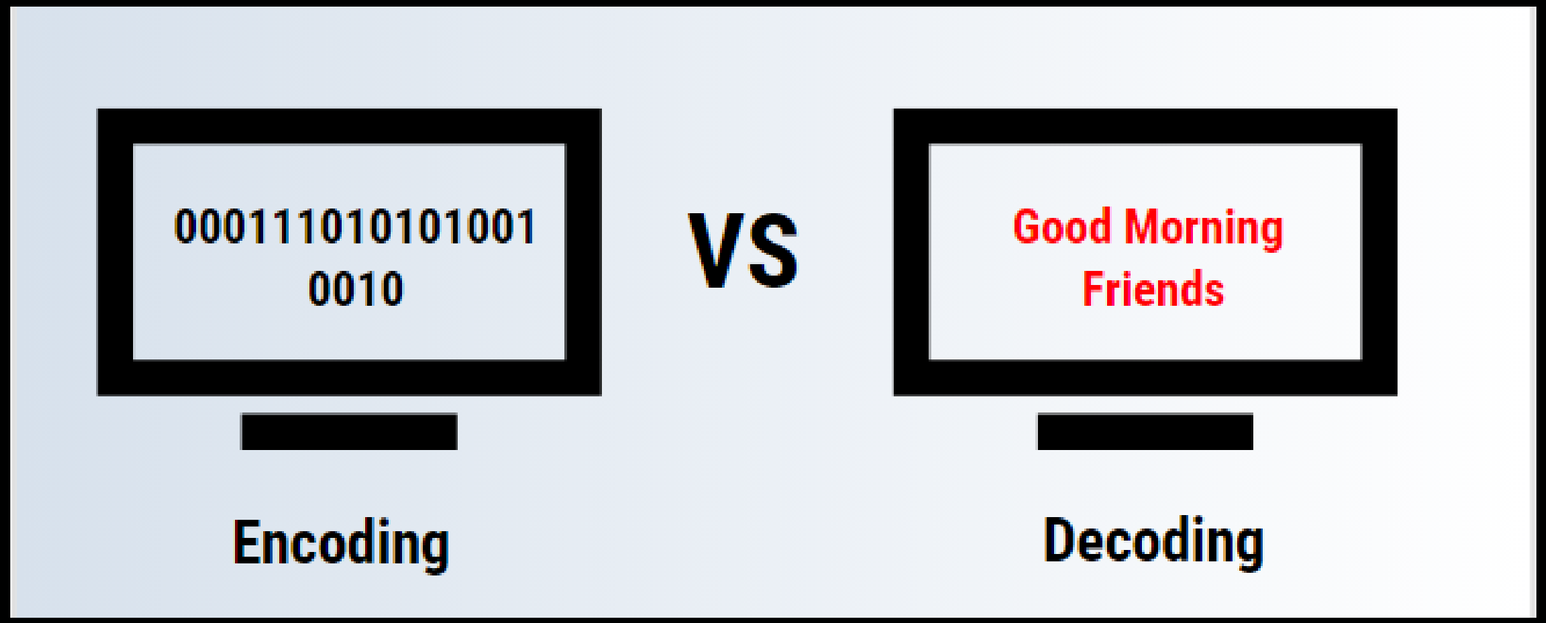
Encoding NOT a form of encryption, just a form of data representation like base64. Immediately reversible, i.e. no key is used.

Encryption is for maintaining data confidentiality and requires the use of a key (kept secret) in order to return to plaintext.

Hashing is for validating the integrity of content by detecting all modification thereof via obvious changes to the hash output.

How Do You Achieve the CIA?		
e.g., Encryption	e.g., Hashing, Digital signatures	e.g., Backups, redundant systems

DECODING/ENCODING



BASE32 & BASE64

- Base32

32-printable characters, It uses uppercase letters A-Z, followed by 2-7 (0 and 1 are skipped due to their similarity with the letters O and I) and the equal sight .

- Base64

In Base64, as the name suggests, there are 64 characters used to encode binary data. These characters are:

- 26 Capital letters [A-Z]
- 26 lower letters [a-z]
- 10 digits [0-9]
- 2 special characters [+ , /]

and of course '='

CTF-1

There are so many different ways of encoding and decoding information nowadays... One of them will work!

Q1RGe0ZsYWdneVdhZ2d5UmFnZ3l9

This CTF is from CTFlearn platform

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CTF-2

I am confused I dont know which base i am in !!!!!!!!!!!!!!

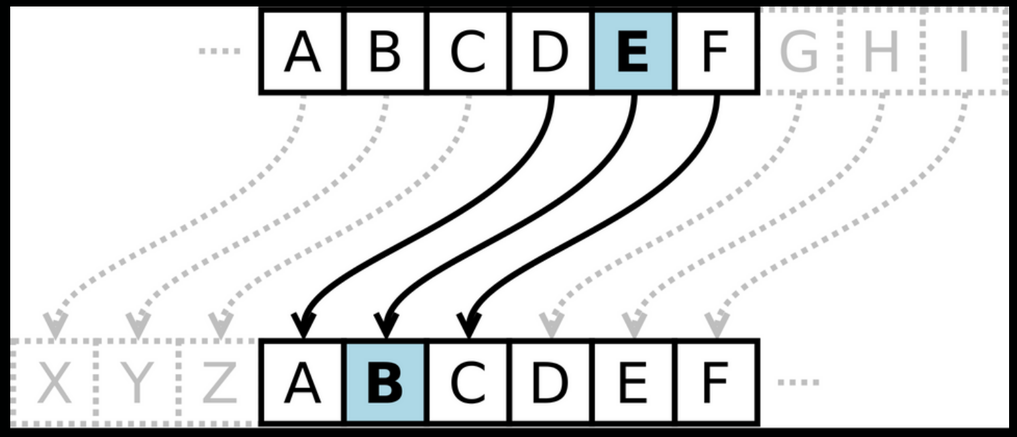
dg4KNUu6vb7CCffN4vmZWda

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CAESAR CIPHER

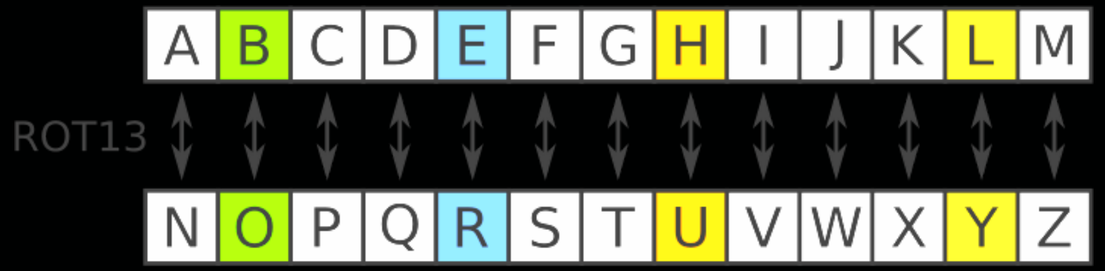
It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet.

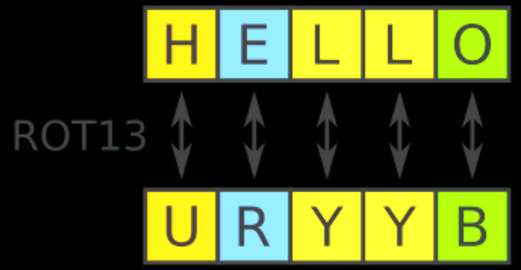
For example, with a **left shift of 3**, D would be replaced by A, E would become B, and so on.



CAESAR CIPHER

ROT13 "rotate by 13 places", is a special case of the Caesar cipher, that replaces a letter with the 13th letter after it in the alphabet.





CTF-3

Cryptography doesn’t have to be complicated,
cvpbPGS{guvf_vf_pelcgb!}

Hint : This CTF is from picoCTF platform

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CTF-4

This is one of the older ciphers in the books, can you decrypt the message?

picoCTF{yjhipvddsdasrpthpgrxewtgdqnjytto}

Hint : Brute Force Is Your Friend

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

OTHERS

- Ascii

067 073 084 123 073 115 084 104 101 077 111 111 111 111 111 111 110 033 125

- base2

01000011 01001001 01010100 01111011 01001001 01110011 01010100 01101000 01100101 01001101 01101111 01101111 01101111 01101111 01101111 01101110 00100001 01111101

- base16

4349547b49735468654d6f6f6f6f6f6f6e217d

CIT{IsTheMooooooon!}

CTF-5

link

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

OTHERS

- Morse code

SequencesOfTwoDifferentSignalDurationCalledDotsAndDashes

In Morse Code :

... . --.- ..- . -. -.-. . ... --- ..-. - .-- --- -.. .. ..-. ..-. . .-. . -. - ... .. --. -. .- .-.. -.. ..- .-. .- - .. --- -. -.-. .- .-.. .-.. . -.. -.. --- - ... .- -. -.. -.. .- ... .

- BrainF**k

using six symbols + , - , < , > , [,] ,

Msg In BrainF**k :

++++[++++>---<]>.++++++.++++++++++.[--->+<]>++++.++++[->++<]>+.+[->+++++<]>+.+[--->+<]>+++.>-[--->+<]>-.[---->+++++<]>-.---.--[--->+<]>-.+[->++<]>.[-->+++<]>++.[--->+<]>----.+.[++>---<]>.

CTF-6

It is That 34sy

```
+[-------->++<]>+.++.-------.[--->+<]>+.[->++++++<]>-.-[--->+<]>--.+[->+++<]>+.++++++++.----
--------.[--->+<]>-.------------.---.[->+++<]>--.-[--->+<]>--.[--->+<]>----.+++[->+++
<]>++.++++++++.+++++.[++>---<]>--.-[-->+++++++<]>.[->+++<]>.--[--->+<]>.+[->+++
<]>+.++++++++.+++++.[->+++<]>++.++++.+++..+.>--[-->+++<]>.
```

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CTF-7

I can be anything

link

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CTF-8

when i was learning the alphabets , by coincidence I was thinking about using an encryption algorithm , so I try to encrypt a message and in result i get this cipher .
"AAABAABAAABAABAABAAABAAABAAAAABABAAAABAABAAABABBABABABBAABAA "

Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

CTF-9

The vignere cipher is a method of encrypting alphabetic text by using a series of interwoven Caesar ciphers based on the letters of a keyword.

I’m not sure what this means, but it was left lying around: blorpy

gwox{RgqssihYspOntqpxs}

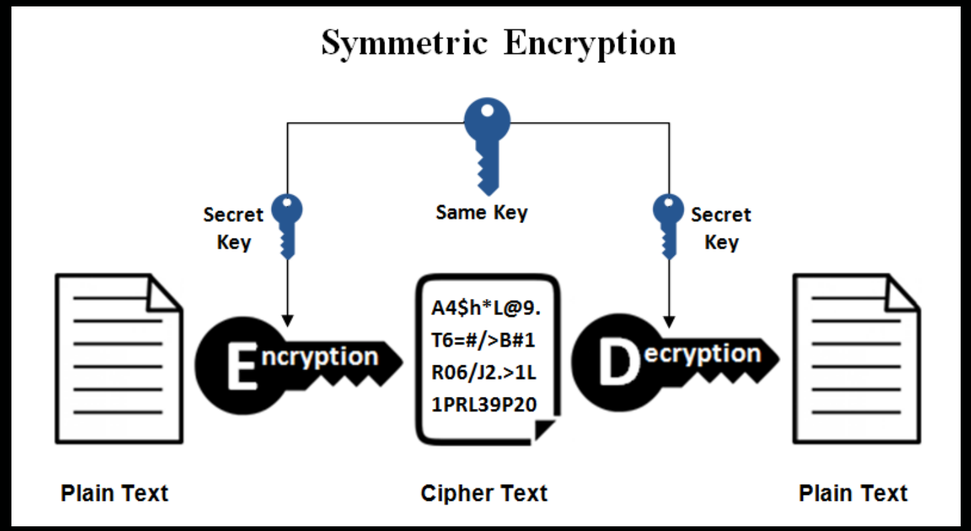
Disclaimer : If you find the flag, Raise your hand and say Done without Spoiling

RESOURCES

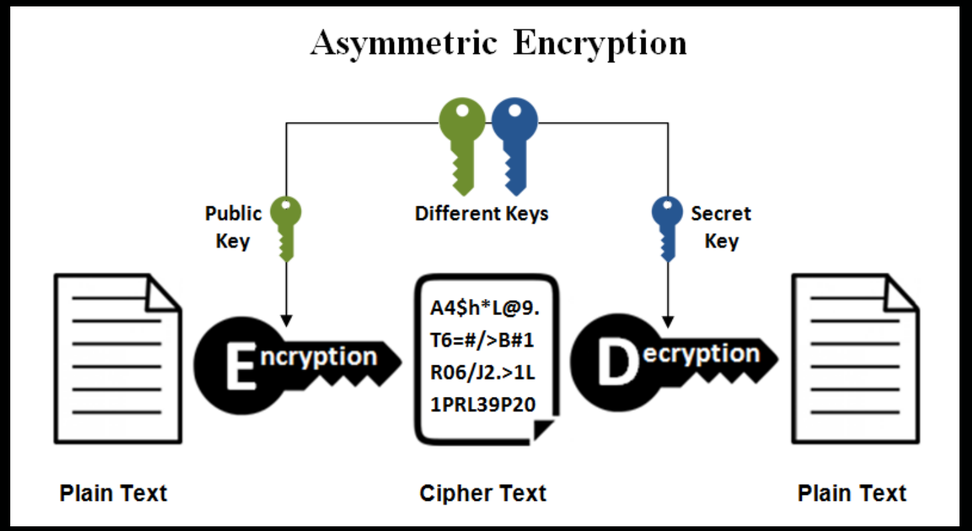
- https://gchq.github.io/CyberChef/
- https://www.dcode.fr/
- https://rot13.com/
- https://morsedecoder.com/
- http://www.unit-conversion.info/texttools/
- https://www.online-toolz.com/
- https://cryptii.com/
- https://mothereff.in/bacon

DECRYPTION/ENCRYPTION

Symmetric Encryption



Asymmetric Encryption



Hybrid Encryption

SYMMETRIC ENCRYPTION ALGORITHMS

- AES (Advanced Encryption Standard)
- DES (Data Encryption Standard)
- IDEA (International Data Encryption Algorithm)
- Blowfish (Drop-in replacement for DES or IDEA)
- RC4 (Rivest Cipher 4)
- RC5 (Rivest Cipher 5)
- RC6 (Rivest Cipher 6)

VIGENERE CYPHER: ENCRYPTION

msg: HELLO WORLD

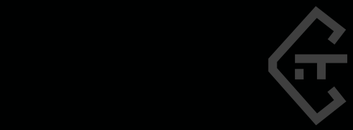
key: TEST

Encryption:

H	E	L	L	O	W	O	R	L	D
8	5	12	12	15	23	15	18	12	4
T	E	S	T	T	E	S	T	T	E
19	4	18	19	19	4	18	19	19	4

msg + key % 26 =

1	9	4	5	8	1	7	11	5	8
A	I	D	E	H	A	G	K	E	H



VIGENERE CYPHER: DECRYPTION

msg: HELLO WORLD
key: TEST

```
if cypher - key >= 0:
msg = cypher - key
if cypher - key < 0:
msg = cypher - key +26
```

Decryption:

A	I	D	E	H	A	G	K	E	H
1	9	4	5	8	1	7	11	5	8
T	E	S	T	T	E	S	T	T	E
19	4	18	19	19	4	18	19	19	4
8	5	12	12	15	23	15	18	12	4
H	E	L	L	O	W	O	R	L	D

VIGENERE CYPHER: BRUTE FORCE

If the key length is small:

Length = 1	==>	26 possibility (caesar cipher)
Length = 2	==>	26**2 = 676 possibility
Length = 3	==>	26**3 = 17576 possibility
Length = 4	==>	26**4 = 456976 possibility
Length = 5	==>	26**5 = 11881376 possibility
Length = 6	==>	26**6 = 308915776 possibility

ASYMMETRIC ENCRYPTION ALGORITHMES

- Rivest Shamir Adleman (RSA)
- the Digital Signature Standard (DSS), which incorporates the Digital Signature Algorithm (DSA)
- Elliptical Curve Cryptography (ECC)
- the Diffie-Hellman exchange method.
- TLS/SSL protocol.

ASYMMETRIC ENCRYPTION: RSA

Components:

p,q : two large primes
n = p.q : modulus
e : exponent, 1 < e < euler(n), e and euler(n) are coprime

$$euler(n) = (p - 1)(q - 1)$$

Public key: n, e

message: m

ASYMMETRIC ENCRYPTION: RSA

Generating m from clear text:

ASCII table

65	41	101	&#65;	A		97	61	141	&#97;	a
66	42	102	&#66;	B		98	62	142	&#98;	b
67	43	103	&#67;	C		99	63	143	&#99;	c
68	44	104	&#68;	D		100	64	144	&#100;	d
69	45	105	&#69;	E		101	65	145	&#101;	e
70	46	106	&#70;	F		102	66	146	&#102;	f
71	47	107	&#71;	G		103	67	147	&#103;	g
72	48	110	&#72;	H		104	68	150	&#104;	h
73	49	111	&#73;	I		105	69	151	&#105;	i
74	4A	112	&#74;	J		106	6A	152	&#106;	j
75	4B	113	&#75;	K		107	6B	153	&#107;	k
76	4C	114	&#76;	L		108	6C	154	&#108;	l
77	4D	115	&#77;	M		109	6D	155	&#109;	m

"HAI" -> m = 72*256**2 + 65*256**1 + 73*256**0= 4735305

"HAI" -> m = 0x484149 = 4669768

TEST - TEST

Convert this to decimal:

Testing RSA !!

In Python:

```
chr(65) == "A"
ord("A") == 65
```

TEST - TEST

Convert this to text:

70858008973140306612395235164449

In Python:

```
chr(65) == "A"
ord("A") == 65
```

ASYMMETRIC ENCRYPTION: RSA

Based on:

Modular Exponentiation: m**e mod n

In Python:

pow(base, exponent, modulus)

ASYMMETRIC ENCRYPTION: RSA

Encryption:

c = (m ** e) % n

Python: c = pow(m,e,n)

Example:

p = 3 , q = 11

n = p * q = 3 * 11 = 33

e = 7

m = 2

c = 2 ** 7 mod 33 = 29

TEST - TEST

Encrypt this text:

Testing RSA !!

Values:

n = 882564595536224140639625987659416029426239230804614613279163

e = 65537

ASYMMETRIC ENCRYPTION: RSA

Private key: d, n

euler(n) = (p - 1)*(q - 1)

e.d = 1 % euler(n)

Python: d = pow(e,-1,euler_n)

d called the modular multiplicative inverse of e

Decryption:

m = (c ** d) % n

Python: m = pow(c,d,n)

Example:

p = 3 , q = 11

n = 33

e = 7

c = 29

euler(n) = (3 - 1)(11 - 1) = 20

d.e = 1 % n

==> d = 3

m = 29 ** 3 % 33 = 2

TEST - TEST

Decrypt this:

325334761016336946446596805334601
271861143244958664513525228

Values:

n = 882564595536224140639625987659416029426239230804614613279163

p = 10292249479429980750803486472l9

e = 65537

CRYPTANALYSIS: RSA ATTACKS

List of RSA attacks:

- Factoring the Public Key
- Guessing d
- Common Modulus
- Low Exponent

...

CRYPTANALYSIS: RSA ATTACKS

- Factoring the Public Key

using factordb.com we can find the primes that construct n

Decrypt this:

792481077342031483370162500948606855670434997623166673723280431109 2

Values:

n = 322691095132643788731511200680744440869890447414186711223387981169 69

e = 65537

HASHING

Definition:

The process of converting any data into same lenght random strings depending on the hash algorithm used

MD5: F82E5C00EFEE8D85A47F7D42853B73DB

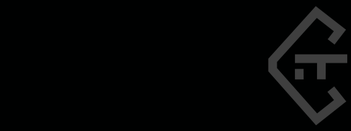
SHA256:

12BDF9B98E7C15BF6E4361AD4663E7D9EDD13D330C8E70AA860F7C8559A28099

SHA512:

4160E7D420B927C65A97F2ACD3ED157E7BFDCA5830547473158F42DD4C431A80737C86F9FC93B338CE7816E2E9CC3411BA02F44337233D088AE406141E8DE960

Why hashing ??



HASHING

Collisions:

two keys can generate the same hash

CRACKING HASHES

Concept:

hashing strings and comparing the hashes with the hashed strings

Tools:

- Hash-identifier
- Hashcat
- John The Ripper

CRACKING HASHES

Crack this hash:

392115d3537e79ed7ac6f5f35b13283c1e8b918d8b93af68f37bd6f19f7d4448

Algorithm used: Keccak-XXX

SOURCES

https://cryptohack.org/
https://cryptopals.com/
https://overthewire.org/wargames/krypton/
https://picoctf.org/

https://github.com/Ganapati/RsaCtfTool