



# Cryptography

## **Introduction to Cryptography...**

Cryptography is the practice and study of techniques for secure communication in the presence of third

parties called adversaries.

From Greek

- Cryptos: secret, hidden
- Graphos: writing
- Cryptography: study (some calls science or art too) of secret writing

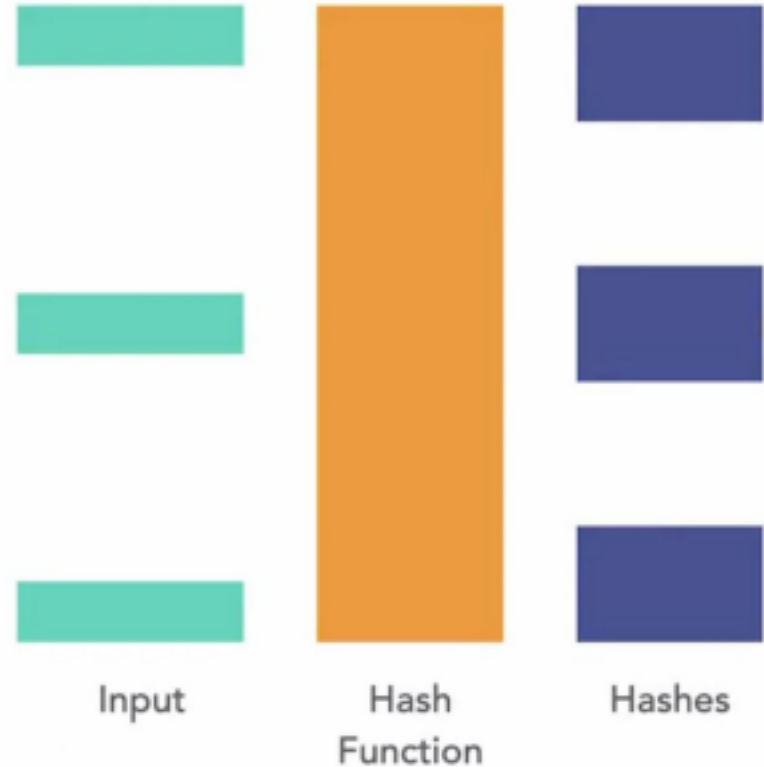
Example: Secure communication using AES encryption for messaging apps like WhatsApp



Concealing information using secret codes began over a thousand years ago.

# Integrity

- To protect integrity, we can use a hash function



Ensures that data is not altered during communication, transmission or storage.

## What is Crypto Currency and why its base is fake?

# Encoding

The purpose of encoding is to represent something in a certain, pre-defined way for transport, without any regard to secrecy or confidentiality. Everything processed by modern-day computers is “encoded” with 1s and 0s, usually grouped into eight-bit chunks called bytes. Some examples of encodings are: character encodings such as ASCII (English text), Braille, Morse code, and UTF-8; and, data encodings such as Base64, Huffman coding, and percent-encoding.

# Obfuscation

The purpose of obfuscation is to make something ambiguous or hard to understand, with the intent of confidentiality, without any auxiliary information. However, in reality, obfuscation merely creates an obstacle (or several obstacles) and thus is not a strong control.

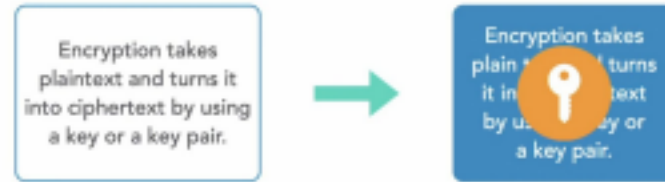
As a side note, this concept is represented in Japanese as *nandoku-ka* (難読化, なんどくか), which aptly means “making something hard to read.”

If you have ever opened the developer tools in a browser to look at Google’s JavaScript code, you will have seen the result of obfuscation—helpful hints like names and structures have been replaced, rendering the code very difficult to understand for humans at a glance, but still keeping its semantics (or utility) intact. It is still possible, with a bit of time and effort, to recreate something resembling the original code.

# Encryption

The purpose of encryption is to prevent the disclosure of information to anyone other than the intended recipient(s). Encryption uses a well-defined, well-known (i.e., public) algorithm with some kind of secret (called a “key”). Symmetric encryption algorithms use one single key for both encryption and decryption. Some examples of symmetric encryption algorithms are AES and Twofish. Asymmetric algorithms, also known as public-key cryptography, use a pair of keys where the one used for encryption cannot be used for decryption. Some examples of asymmetric encryption algorithms are [ECDSA](#) and [RSA](#).

## Encrypting Data



- 1- Base64 Encode/Decode a message
  - 2- Reverse the message text
  - 3- Shift every alphabet by 5
- Any other you can suggest?

## Bank of America Data loss

In December of 2004, Bank of America employees backed-up data and sent backup tapes containing names, addresses, bank account numbers, and Social Security numbers of 1.2 million government workers

enrolled in charge-card (Credit Cards) accounts. Data was not encrypted. The tapes never arrived and indeed have never been found. Sadly, this method of backing up and shipping data is all too common.

[rollcall.com/2005/03/01/staff-data-lost-by-bank-of-america](http://rollcall.com/2005/03/01/staff-data-lost-by-bank-of-america)

# Types of Cryptography

## Symmetric Cryptography

In symmetric or secret key cryptography, there is only one secret key between the parties involved, typically referred to as the "symmetric key" or "shared key."

## Asymmetric Cryptography

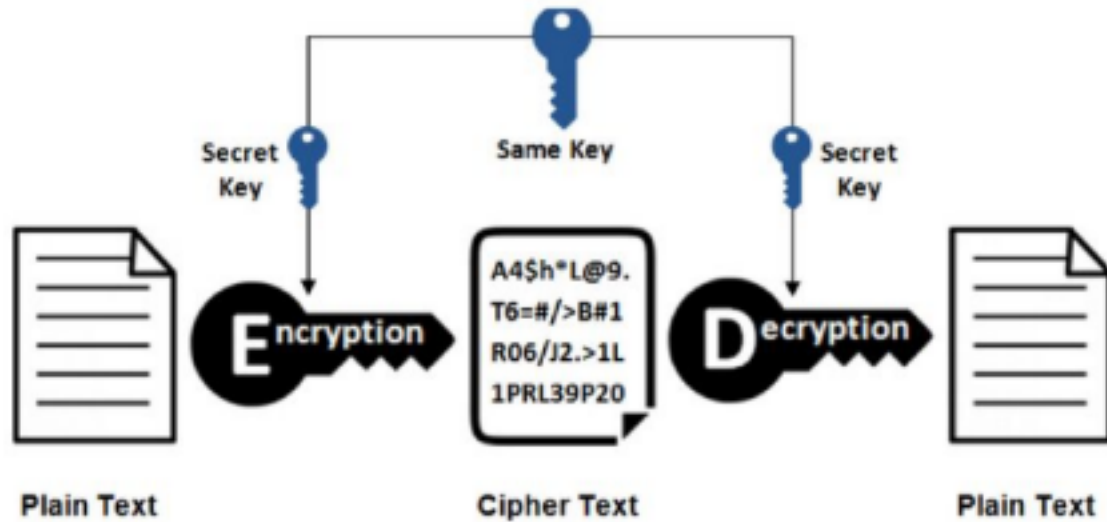
Asymmetric or public key cryptography uses a pair of keys for each party: a public key and a private key. The public key is widely known and can be freely shared, while the private key is kept secret.

## Hash Functions

Hash functions do not require a key. Instead, they use mathematical algorithms to convert messages of any arbitrary length into a fixed-length output, known as a hash value. Hash functions are designed to be

one-way, meaning the original input cannot be derived from the output. Unique fingerprint of data ensures its integrity with no chance of recovery.

## **Symmetric Encryption Explanation**



## Symmetric Cryptography Challenges

- two requirements for secure use of symmetric encryption:
  - strong encryption and decryption algorithms
  - a secret key known only to sender / receiver
$$Y = E_K(X) \quad \text{or} \quad Y = E(K, X)$$
$$X = D_K(Y) \quad \text{or} \quad X = D(K, Y)$$
- assume encryption algorithm is known
- a secure channel is needed to distribute key

## Basic Terminologies

**plaintext** - the original message

**ciphertext** - the coded message

**cipher** - algorithm for transforming plaintext to ciphertext

**key** - info used in cipher known only to sender/receiver

**encipher (encrypt)** - converting plaintext to ciphertext

**decipher (decrypt)** - recovering plaintext from ciphertext

**cryptography** - study of encryption principles/methods

**cryptanalysis (codebreaking)** - the study of principles/methods of deciphering ciphertext without knowing key

# Kerckhoffs' principles

“The security of a cipher must not depend on anything that cannot be easily changed”

“The opponent is not to be underestimated. In particular, the opponent knows the encryption and decryption algorithms. So the strength of a cipher system depends on keeping the key information secret, not the algorithm”

Auguste Kerckhoff, 1883

# Monoalphabetic ciphers

- Monoalphabetic ciphers
    - shuffle the letters arbitrarily based on a 26 letters long key
- Plain:    abcdefghijklmnopqrstuvwxyz  
Cipher:  DKVQFIBJWPESCXHTMYAUOLRGZN
- Plaintext:    ifwewishtoreplaceletters  
Ciphertext:  WIRFRWAJUH YFTSDVFSFUUFYA

See the example on pages 93, 94, 95 and 96 of text book (Cryptography and Network Security: Principles and Practice)

## Playfair Cipher

The best-known multiple-letter encryption cipher is the Playfair, which treats digrams in the plaintext as single units and translates these units into ciphertext digrams.

M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

## Playfair Encryption

1. **Choose a keyword:** Select a keyword (usually a word or phrase) that will be used to generate the key for the Playfair cipher. For example, let's use the keyword "MONARCHY."
2. **Create the key table:** Create a 5x5 grid (also called a key table) using the letters of the keyword, without repeating any letters. Fill in the remaining empty spaces with the remaining letters of the alphabet, excluding any letters from the keyword. Make sure to keep the letters in their natural order (left to right, top to bottom). Above is an example of a key table using the keyword "MONARCHY":
3. **Prepare the message:** Remove any spaces and punctuation from the plaintext message and convert it to uppercase letters. For example, "HELLO WORLD" becomes "HELLOWORLD."
4. **Break the message into pairs:** Divide the prepared message into pairs of two letters. If there is an odd number of

letters or similar letters in a pair, add a placeholder (usually an "X") to make pairs. For example, "HELLOWORLD" becomes "HE LX LO WO RL DX."

**5. Encrypt the pairs:** For each pair of letters, apply the following rules:

- a. If both letters are in the same row of the key table, replace them with the letters to their right (loop back to the beginning of the row if necessary).
- b. If both letters are in the same column of the key table, replace them with the letters below (loop back to the top of the column if necessary).
- c. If the letters are in different rows and columns, form a rectangle with the two letters and replace them with the letters at the opposite corners of the rectangle.

## Playfair Encryption

HE → CF

LX → SU

LO → PM

WO → VN

RL → MT

DX → BZ

If plaintext is HELLO WORLD then ciphertext will be **cfsupmvnmtbz** using **MONARHY** as the key

1- Can you get back hello world using Playfair decryption?

2- Can you encrypt and decrypt “SANA SAFINAZ”?

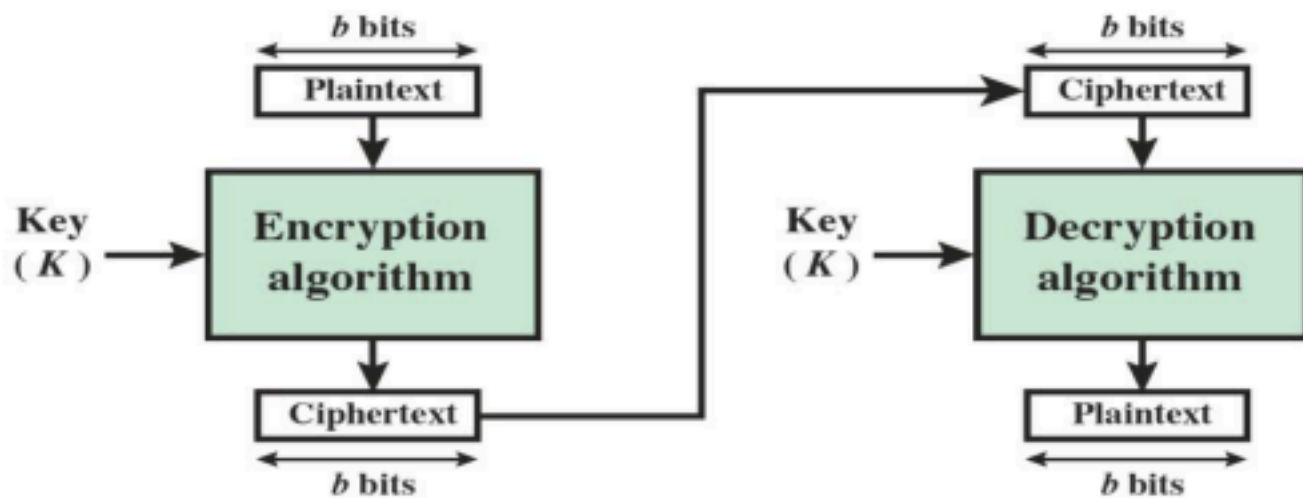
## Playfair Decryption

1. Prepare the ciphertext by removing spaces and punctuation, converting it to uppercase.
2. Break the ciphertext into pairs of letters.
3. Decrypt each pair using the reverse of the encryption rules, using the key table.

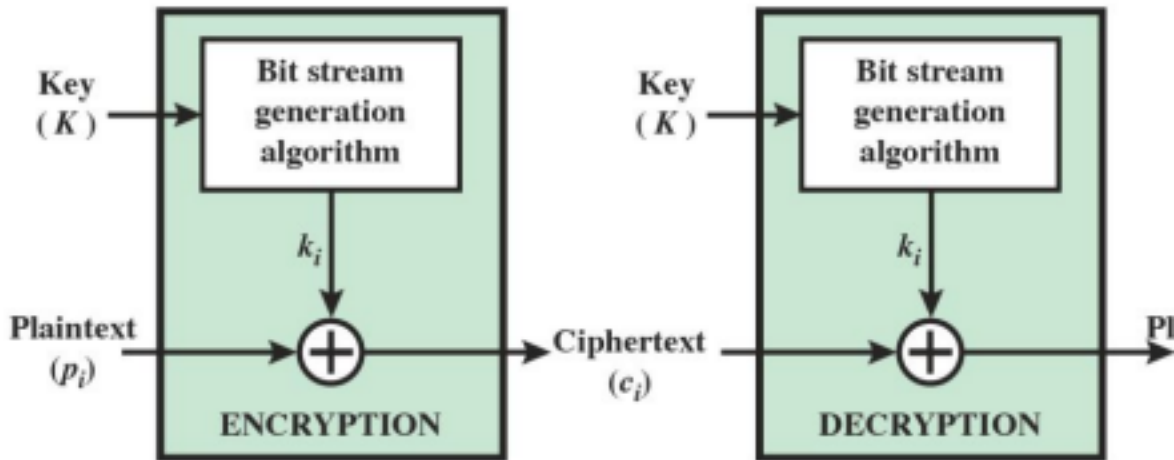
Remember that the Playfair cipher is not very secure by modern standards and can be easily cracked with computer algorithms. It's mainly a historical curiosity rather than a secure encryption method.

# Modern Cryptography

- Block ciphers vs. Stream Ciphers
- Block ciphers operate on a block of data
  - entire block must be available before processing



- Stream ciphers process messages one bit or byte at a time when en/decrypting
  - need not wait the entire block



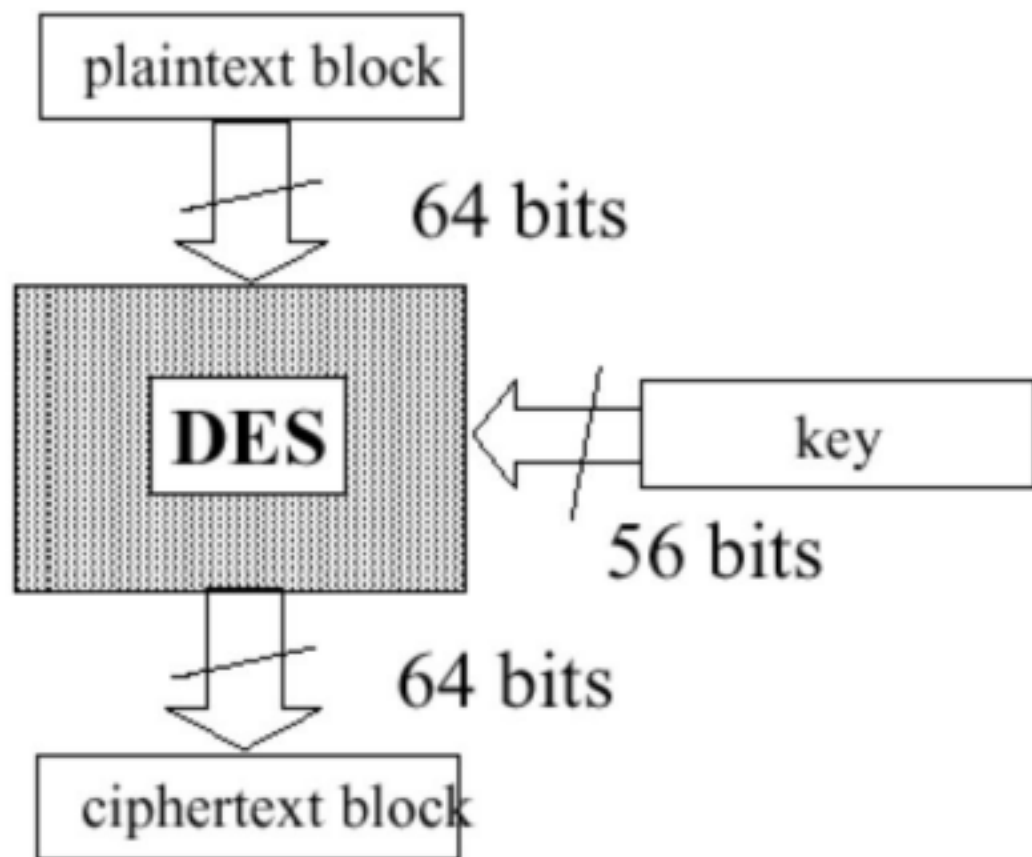
(a) Stream Cipher Using Algorithmic Bit Stream Generator

- Most ciphers are block ciphers

## DES (Data Encryption Standard)

- Most widely used block cipher in world

- Adopted in 1977 by NBS (now NIST)
- Encrypts 64-bit data using 56-bit key
- Had widespread use
- There has been considerable controversy over its security



# Other Important Symmetric Ciphers

- AES (Rjindael)
- 3DES (Triple DES)
- Blowfish
- RC5
- IDEA
- RC4

## AES (Advanced Encryption Standard)

It supports three different block sizes of 128, 192 and 256 bits and so the key lengths of 128, 192 and 256 bits respectively.

Consists of multiple rounds of processing

- 10 rounds for 128 bit key
- 12 rounds for 192 bit key
- 14 rounds for 256 bit key

## AES Requirements

- Private key symmetric block cipher ●

128-bit data (block size)

- 128/192/256-bit keys

- Stronger & faster than Triple-DES ●

Active life of 20-30 years

- Provide full specification and design details

# Asymmetric Cryptography

**Encryption and Decryption:** When someone wants to send an encrypted message to a recipient, they use the recipient's public key to encrypt it. The recipient uses their private key to decrypt the message. The reverse process is also possible, where the sender can digitally sign a message with their private key, and the recipient can verify it using the sender's public key.

**Security:** Asymmetric cryptography provides a higher level of security than symmetric cryptography, mainly because the private key is not shared or exposed during the encryption or decryption process. This makes it less vulnerable to key distribution issues.

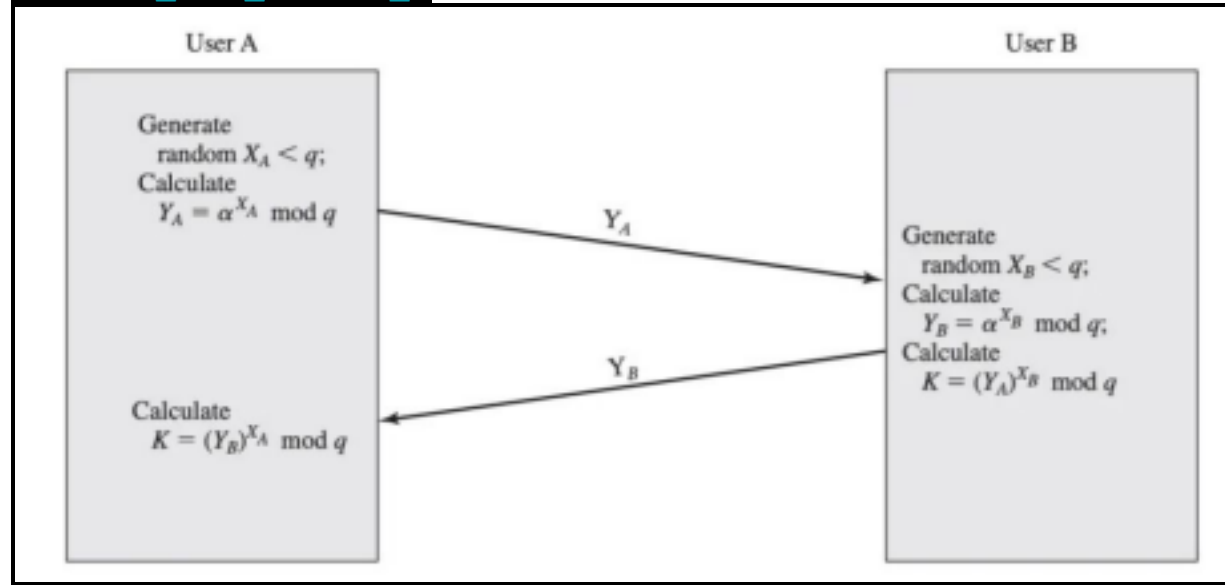
**Key Management:** Asymmetric cryptography requires careful management of key pairs, which can be computationally intensive. However, it simplifies key distribution since public keys can be openly shared.

**Examples:** Common asymmetric encryption algorithms include RSA (Rivest–Shamir–Adleman), DSA (Digital Signature Algorithm), and ECC (Elliptic Curve Cryptography) etc.

## Diffie-Hellman key exchange protocol

Developed by Whitfield Diffie and Martin Hellman in 1976. This protocol aimed to allow sharing of the secret key and encrypting the message depending upon the discrete logarithm computational capacity. Two known public parameters prime number ( $q$ ) and primitive root of ( $q$ ) which is alpha ( $\alpha$ ) are defined before exchanging the secret key between the two parties.

### Primitive\_root\_modulo\_n



# Diffie-Hellman key exchange example

★ Question: What if  $g^x = 9$  and  $q = 23$

★ Question: What if  $g^x = 327$  and  $q = 919$

★ Do you see any issues in DH process?

★ Book Chapter 9

★ What is RSA?

## Cryptanalysis

### Ciphertext-Only Attacks

- Attacker has: Encrypted ciphertexts but no plaintext. •

Goal: Recover plaintext or key.

### Known-Plaintext Attack

- Attacker has: Access to both plaintext and corresponding ciphertext. •

Goal: Determine the encryption key or decrypt future ciphertexts.

### Chosen-Plaintext Attack (CPA)

- Attacker can: Choose plaintexts and obtain their encrypted versions. •

Goal: Derive key information or break encryption.

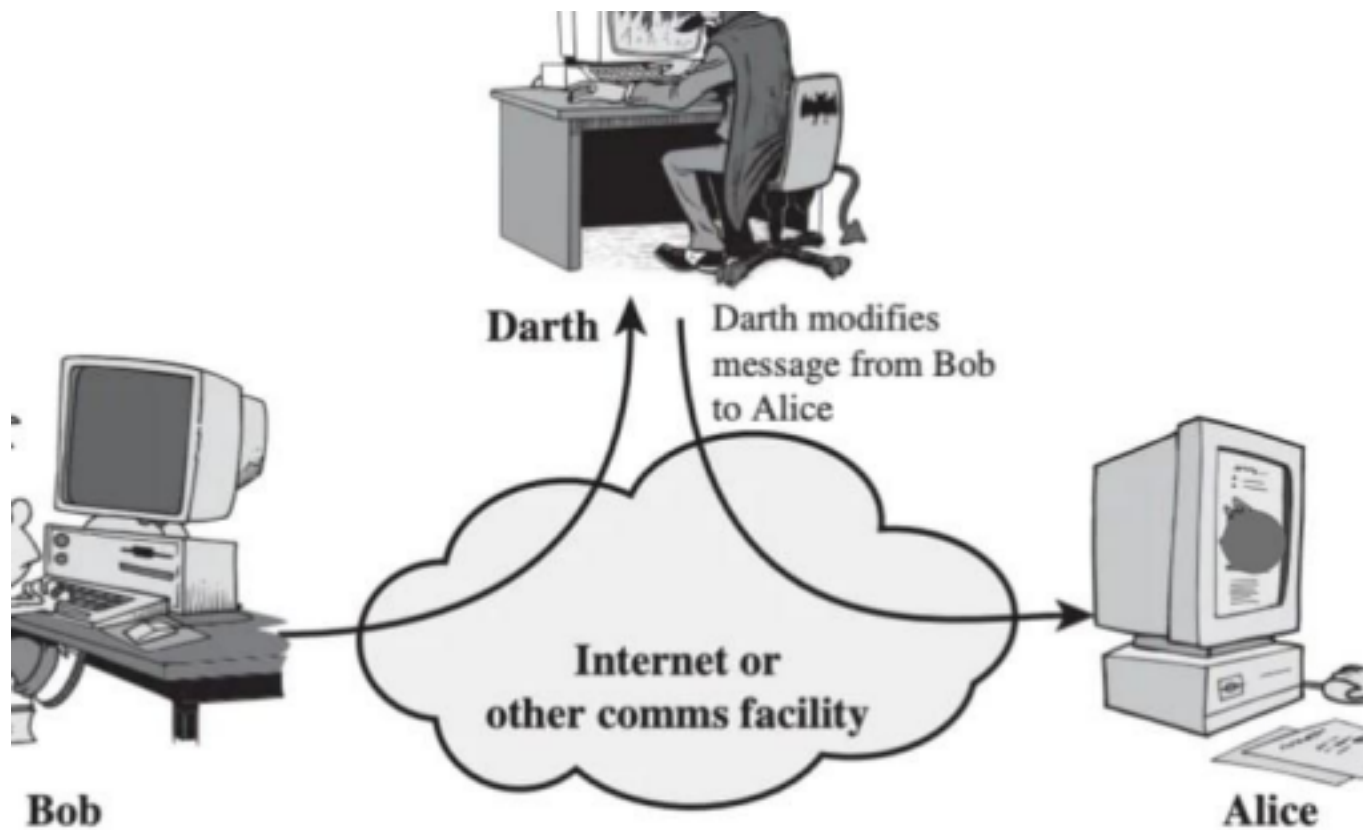
### **Brute-Force Attack**

- Attacker tries: Every possible key until finding the correct one

### **Man-in-the-Middle-Attacks**

- Attacker aims to intercept and manipulate the key exchange process

### **Man-in-the-Middle-Attack Steps**



# Brute Force Search

- Simply try every key
- On average, half of the key space is searched until an intelligible translation is found

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at $10^9$ decryptions/s	Time Required at $10^{13}$ decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	$2^{55}$ ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	$2^{127}$ ns = $5.3 \times 10^{21}$ years	$5.3 \times 10^{17}$ years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	$2^{167}$ ns = $5.8 \times 10^{33}$ years	$5.8 \times 10^{29}$ years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	$2^{191}$ ns = $9.8 \times 10^{40}$ years	$9.8 \times 10^{36}$ years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	$2^{255}$ ns = $1.8 \times 10^{60}$ years	$1.8 \times 10^{56}$ years
26 characters (permutation)	Monoalphabetic	$26! = 4 \times 10^{26}$	$2 \times 10^{26}$ ns = $6.3 \times 10^9$ years	$6.3 \times 10^6$ years

# Activity

- 1- <https://cryptohack.org/challenges>
- 2- <https://overthewire.org/wargames/krypton/krypton1.html>