ADVANCED ENCYPTION STANDARD( AES)

* Advanced Encryption Standard (**AES**) is **symmetric** encryption algorithm.
* AES is an **iterative** rather than Feistel cipher
* AES is widely used today as it is much **stronger than DES** and **triple DES** despite being harder to implement.
  + A replacement for DES was needed as its key size was too small
  + Triple DES was designed to overcome this drawback but it was found slow.
* Advanced Encryption Standard (AES) is a highly trusted **encryption algorithm** used to secure data by converting it into an unreadable format without the proper key
* The features of AES:
  + **Symmetric key symmetric block cipher**
  + **128-bit data**( **encrypt data in blocks of 128 bits each**), **128/192/256-bit keys length**
  + Stronger and faster than Triple-DES
  + Provide full specification and design details
  + Software implementable in C and Java
* This **data security** measure is efficient and widely implemented in securing **internet communication**, protecting **sensitive data**, and **encrypting files.**
* AES relies on the **substitution-permutation** **network principle**, which is performed using a series of linked operations that involve replacing inputs by specific outputs( substitutions) and others involve shuffling bits around( permutation).
* AES is also among the fastest symmetric encryption algorithms, making it more practical to use at scale in **real-life applications**.

# KEY CONCEPTS

* **Symmetric Encryption**: Uses the same key for both encryption and decryption. Keep this key secret!
* **Cipher Block Chaining( CBC) mode**: Each block of plaintext is XORed with the previous ciphertext block before being encrypted. This mode requires an **Initialization Vector** (IV) for the first block.
* **Padding**: AES requires the data length to be a multiple of 16 bytes. Padding is added to the plaintext to meet this requirement.
* **Initialization Vector (IV)**: A random value that ensures different ciphertexts even if the same plaintext and key are used multiple times.

# OPERATION OF AES

* AES performs operations on bytes of data rather than in bits. Since the block size is 128 bits, the cipher processes 128 bits (or 16 bytes) of the input data at a time.
* These **16 bytes** are arranged in **four columns** and **four rows** for processing as a matrix.

A close-up of a number

Description automatically generated

* Unlike DES, the number of rounds in AES is variable and depends on the length of the key:
  + 128-bit key – 10 rounds
  + 192-bit key – 12 rounds
  + 256-bit key – 14 rounds

A diagram of a key exchange

Description automatically generated

# ENCRYPTION

* Each of these steps operates on a block of exactly 16 bytes called the **state**, which is represented as a 4x4 array
* AES considers each block as a 16-byte (4 byte x 4 byte = 128 ) grid in a column-major arrangement.

A close-up of a number

Description automatically generated

* Each round comprises of 4 steps:
  + SubBytes
  + ShiftRows
  + MixColumns
  + Add Round Key



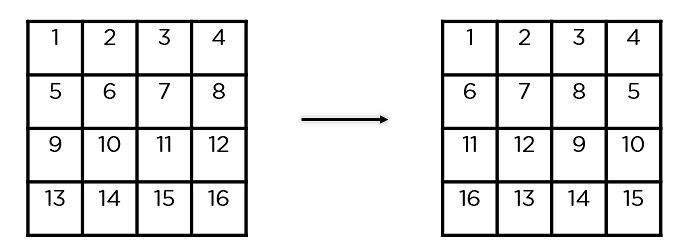
* The last round doesn’t have the MixColumns round.
* The SubBytes does the substitution and ShiftRows and MixColumns perform the permutation in the algorithm.

## **Bytes Substitution( Sub bytes)**

* **This step implements the substitution.**
* In this step, each byte is substituted by another byte.
* It is performed using a lookup table also called the [S-box](https://www.geeksforgeeks.org/what-is-s-box-substitution/).
* This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte.
* The result of this step is a 16-byte (4 x 4 ) matrix like before.

## **Shift Rows**

* **This step implements the permutation.**
* Each of the four rows of the matrix is shifted to the left.
* Any entries that ‘fall off’ are re-inserted on the right side of row.
* Each row is shifted a particular number of times.
  + The 1st row is not shifted.
  + The 2nd row is shifted once to the left.
  + The 3rd row is shifted twice to the left.
  + The 4th row is shifted thrice to the left.



## **Mix Columns**

* **This step implements the permutation**
* This step is a matrix multiplication.
* Each column of four bytes is now transformed using a special mathematical function.
* This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column.
* The result is another new matrix consisting of 16 new bytes.

A diagram of a number and a number

Description automatically generated with medium confidence

* This step is not performed in the last round

## **Add Round Keys**

* The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key
* If this is the last round then the output is the ciphertext
* Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

A black and white diagram of numbers and symbols

Description automatically generated