**AES:**

What is the Advanced Encryption Standard?

In the 1990’s the US Government wanted to standardize a cryptographic algorithm, which was to be used universally by then. It was to be called as the Advanced Encryption Standard(AES). Many proposals were submitted and after a lot of debate, an algorithm called as Rijndeal was accepted.Rijndeal was developed by Joan Daemen and Vincent Rijmen (both from Belgium). The name Rijndeal was also based on their surnames (Rijmen and Daemen).

**The AES Encryption algorithm (also known as the Rijndael algorithm) is a symmetric block cipher algorithm with a block/chunk size of 128 bits. It converts these individual blocks using keys of 128, 192, and 256 bits. Once it encrypts these blocks, it joins them together to form the ciphertext.**

**It is based on a substitution-permutation network, also known as an SP network. It consists of a series of linked operations, including replacing inputs with specific outputs (substitutions) and others involving bit shuffling (permutations).**

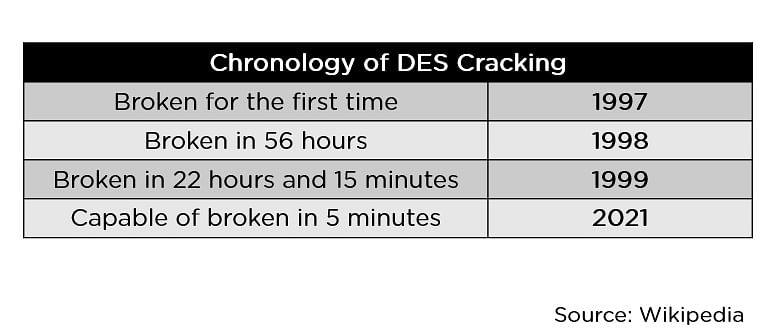
So far, the Advanced Encryption Standard has achieved the targets placed during its inception. And it has a long way to grow.

Why Was the AES Encryption Algorithm necessary?

**When the AES Encryption algorithm succeeded the Data Encryption Standard as the global standard for encryption algorithms in 2001, it fixed many shortcomings of its predecessor. It was seen as the future for encryption in daily life applications.**

The 56-bit keys of DES were no longer considered safe against attacks based on exhaustive key searches and the 64-bits blocks were also considered as weak. AES was to be based on 128-bit blocks, with 128-bit keys.

Going by today’s computational standards, breaking into the [DES algorithm](https://www.simplilearn.com/what-is-des-article) became easier and faster with every year, as seen in the image below.



## What are the Features of AES?

**This robust security algorithm may be implemented in both hardware and software**. It is resilient against hacking attempts, for its higher-length key sizes (128, 192, and 256 bits). It is an open source solution.

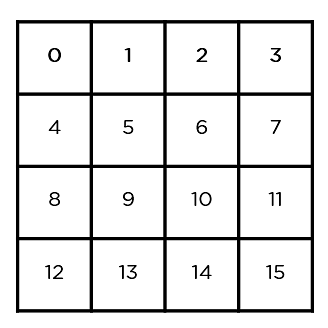
1. **Security.** AES brings additional security because it uses a key expansion process in which the initial key is used to come up with a series of new keys called round keys. These round keys are generated over multiple rounds of modification, each of which makes it harder to break the encryption
2. **Implementation.** Factors to be considered included the algorithm's flexibility, suitability for hardware or software implementation, and overall simplicity.
3. **SP Network**: It works on an SP network structure rather than a Feistel cipher structure, as seen in the case of the DES algorithm.

## 

## How Does AES Work?

AES is a symmetric key cipher. This means the same secret key is used for both encryption and decryption, and both the sender and receiver of the data need a copy of the key. By contrast, asymmetric key systems use a different key for each of the two processes. Asymmetric keys are best for external file transfers, whereas symmetric keys are better suited to internal encryption. The advantage of symmetric systems like AES is their speed. Because a symmetric key algorithm requires less computational power than an asymmetric one, it’s faster and more efficient to run.

To understand the way AES works, we first need to learn how it transmits information between multiple steps. Since a single block is 16 bytes, a 4x4 matrix holds the data in a single block, with each cell holding a single byte of information.



The matrix shown in the image above is known as a state array. Similarly, the key being used initially is expanded into (n+1) keys, with n being the number of rounds to be followed in the encryption process. So for a 128-bit key, the number of rounds is 10, with no. of keys to be generated being 10+1, which is a total of 11 keys.

**Algorithm:**

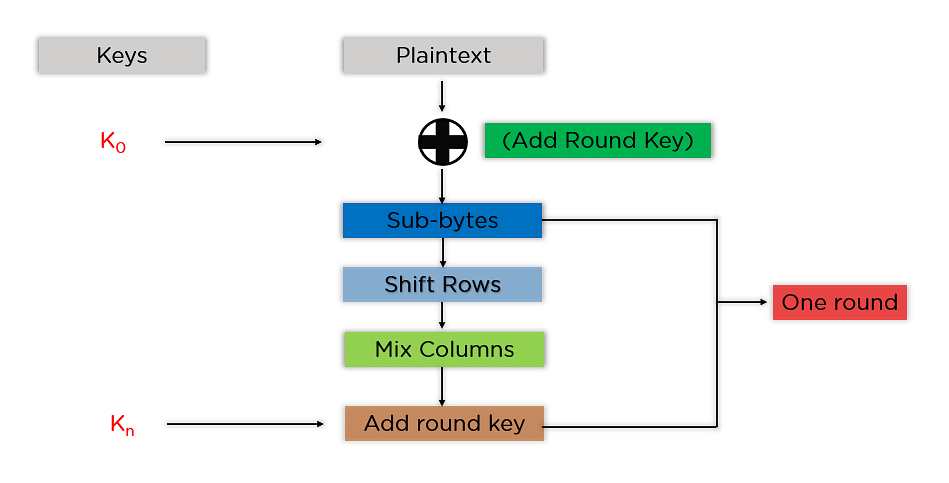
**(i)** Do the following one-time initialization processes:

1. Expand th 16-byte key to get the, actual key block to be used.
2. Do one time initialization of the 16-byte plain text block(calledd as State).
3. XOR the state with the key block.

(ii) For each round, do the following:

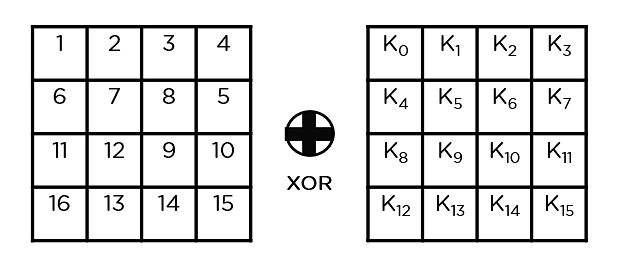
1. Apply S-box to each of the plain text bytes.
2. Rotate row k of the plain text block(i,e.state) by k bytes.
3. Perform a mix columns operation.
4. XOR the state with the key block.

The steps to be followed in AES are in the below image.

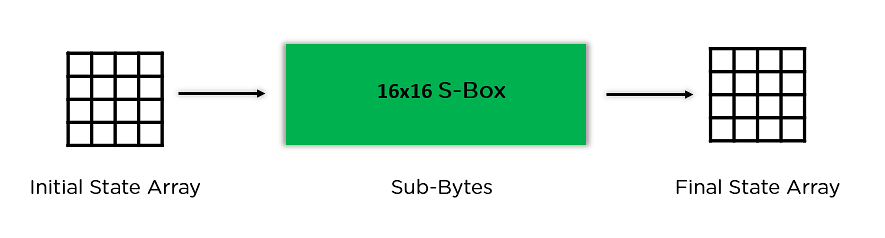


The mentioned steps are to be followed for every block sequentially. Upon successfully encrypting the individual blocks, it joins them together to form the final ciphertext. The steps are as follows.

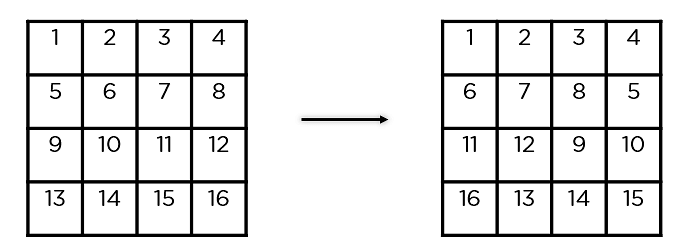
* **Add Round Key:** We pass the block data stored in the state array through an XOR function with the first key generated (K0). It passes the resultant state array on as input to the next step.



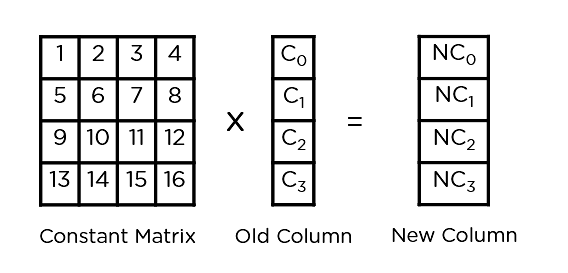
* **Sub-Bytes:** In this step, it converts each byte of the state array into hexadecimal, divided into two equal parts. These parts are the rows and columns, mapped with a substitution box (S-Box) to generate new values for the final state array.



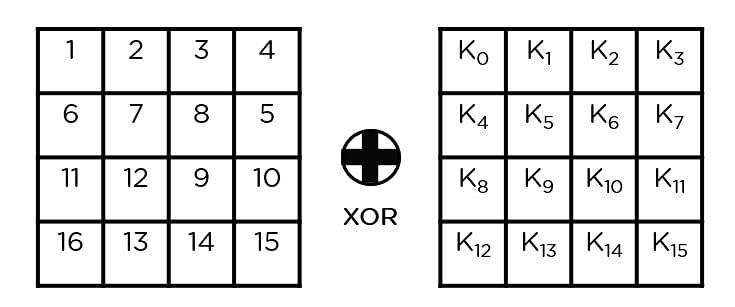
* **Shift Rows:** It swaps the row elements among each other. It skips the first row. It shifts the elements in the second row, one position to the left. It also shifts the elements from the third row two consecutive positions to the left, and it shifts the last row three positions to the left.



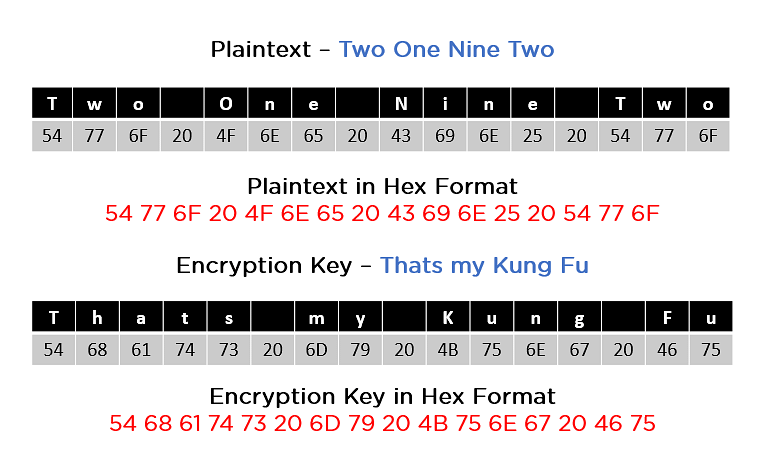
* **Mix Columns:** It multiplies a constant matrix with each column in the state array to get a new column for the subsequent state array. Once all the columns are multiplied with the same constant matrix, you get your state array for the next step. This particular step is not to be done in the last round.



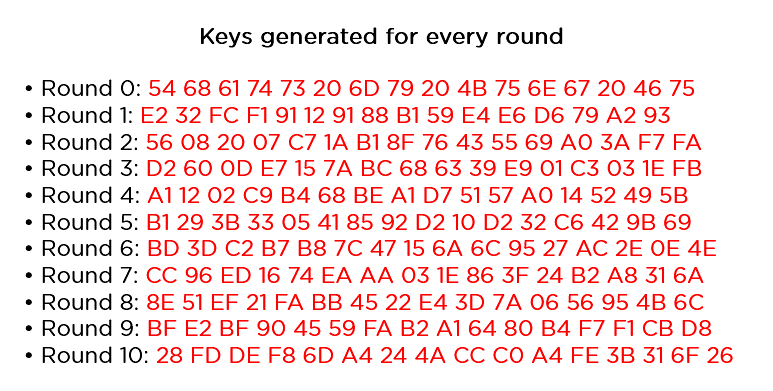
* **Add Round Key:** The respective key for the round is XOR’d with the state array is obtained in the previous step. If this is the last round, the resultant state array becomes the ciphertext for the specific block; else, it passes as the new state array input for the next round.



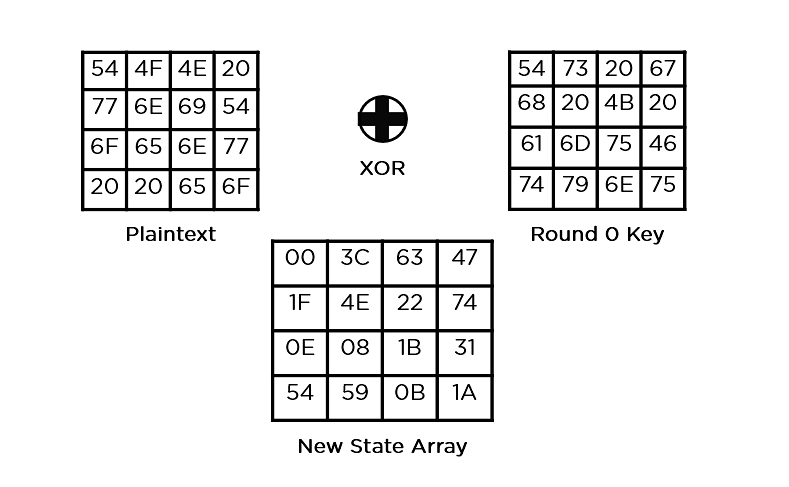
Now that we understand the basic steps needed to go through the encryption procedure. Let’s understand this example to follow along.

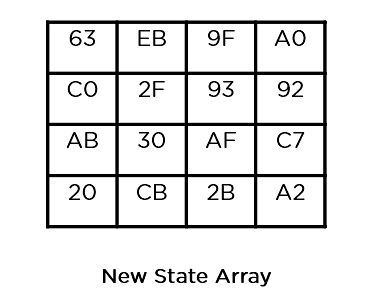


As we can see in the image above, the plaintext and encryption keys convert to hex format before the operations begin. Accordingly, we can generate the keys for the next ten rounds, as you can see below.

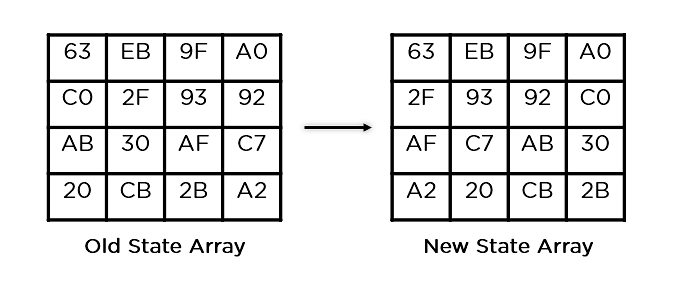


We need to follow the same steps explained above, sequentially extracting the state array and passing it off as input to the next round. The steps are as follows:

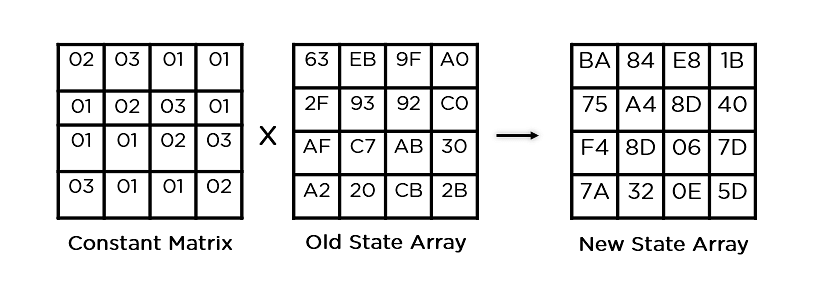
* **Add Round Key:**
* **Sub-Bytes**: It passes the elements through a 16x16 S-Box to get a completely new state array.



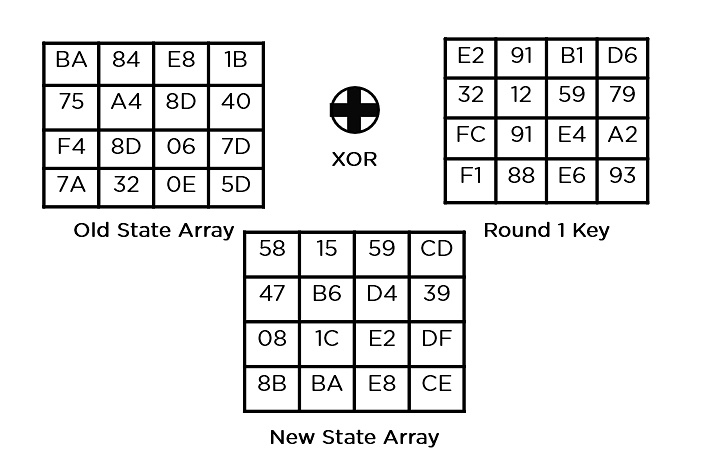
* **Shift Rows:**



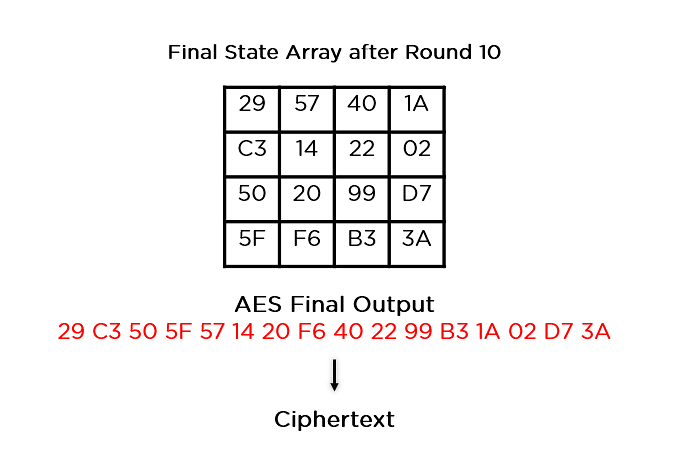
* **Mix Columns:**



* **Add Round Key:**



This state array is now the final ciphertext for this particular round. This becomes the input for the next round. Depending on the key length, we repeat the above steps until we complete round 10, after which you receive the final ciphertext.



Now that you understand how AES works, go through some of the applications of this encryption algorithm.

## What Are the Applications of AES?applications-aes_encryption

The applications of the AES Encryption algorithm are as follows:

1. Wireless Security: Wireless networks are secured using the Advanced Encryption Standard to authenticate routers and clients. WiFi networks have firmware software and complete security systems based on this algorithm and are now in everyday use.
2. Encrypted Browsing: AES plays a huge role in securing website server authentication from both client and server end. With both [symmetric](https://www.simplilearn.com/tutorials/cryptography-tutorial/symmetric-encryption) and [asymmetric encryption](https://www.simplilearn.com/tutorials/cryptography-tutorial/asymmetric-encryption) being used, this algorithm helps in SSL/TLS encryption protocols to always browse with the utmost security and privacy.
3. General File Encryption: Apart from corporate necessities, AES is also used to transfer files between associates in an encrypted format. The encrypted information can extend to chat messages, family pictures, legal documents, etc.
4. Processor Security: Many processor manufacturers enable hardware-level encryption using the likes of AES encryption to bolster security and prevent meltdown failures, among other low-profile risks.

Now that you learned about the applications of AES encryption, take a look at its upgrades over its predecessor, the DES encryption algorithm.

What are the applications of AES?

**Furthermore, AES is often included in commercial based products, including but limited to:**

* Wi-Fi (can be used as part of WPA2)
* Mobile apps (such as WhatsApp and LastPass)
* Native Processor Support.
* Libraries in many software development languages.
* VPN Implementations.
* Operating system components such as file systems.

## **How secure is AES 256 encryption?**

The National Institute of Standards and Technology selected three “flavors” of AES: 128-bit, 192-bit, and 256-bit. Each type uses 128-bit blocks. The difference lies in the length of the key. As the longest, the 256-bit key provides the strongest level of encryption. With a 256-bit key, a hacker would need to try 2256 different combinations to ensure the right one is included. This number is astronomically large, landing at 78 digits total. It is exponentially greater than the number of atoms in the observable universe. Understandably, the US government requires 128- or 256-bit encryption for sensitive data.

The three AES varieties are also distinguished by the number of rounds of encryption. AES 128 uses 10 rounds, AES 192 uses 12 rounds, and AES 256 uses 14 rounds. The more rounds, the more complex the encryption, making AES 256 the most secure AES implementation. It should be noted that with a longer key and more rounds comes higher performance requirements. AES 256 uses 40% more system resources than AES 192, and is therefore best suited to high sensitivity environments where security is more important than speed.