A Report on

Cryptoware: AES-256 (CBC mode)

(Mini-Project of CSE1007-Introduction to Cryptography)

Submitted by

koganti saranya

(Registration-19BCN7187)

on

15 November 2020

School of Computer Science and Engineering

VIT-AP University, Andhra Pradesh

Abstract

The Advanced Encryption Standard (AES) is a symmetric block cipher chosen by the U.S. government to protect classified information. AES is implemented in software and hardware throughout the world to encrypt sensitive data. It is essential for government computer security, cybersecurity and electronic data protection. Cipher Block Chaining (CBC) mode is a block mode of DES that XORs the previous encrypted block of ciphertext to the next block of plaintext to be encrypted. The first encrypted block is an initialization vector that contains random data. This "chaining" destroys patterns.

1 Introduction

The Advanced Encryption Standard (AES), also known by its original name Rijndael is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001. AES is a subset of the Rijndael block cipher developed by two Belgian cryptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal to NIST during the AES selection process.Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.AES has been adopted by the U.S. government and is now used worldwide. It supersedes the Data Encryption Standard (DES), which was published in 1977. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data.

1.1 About

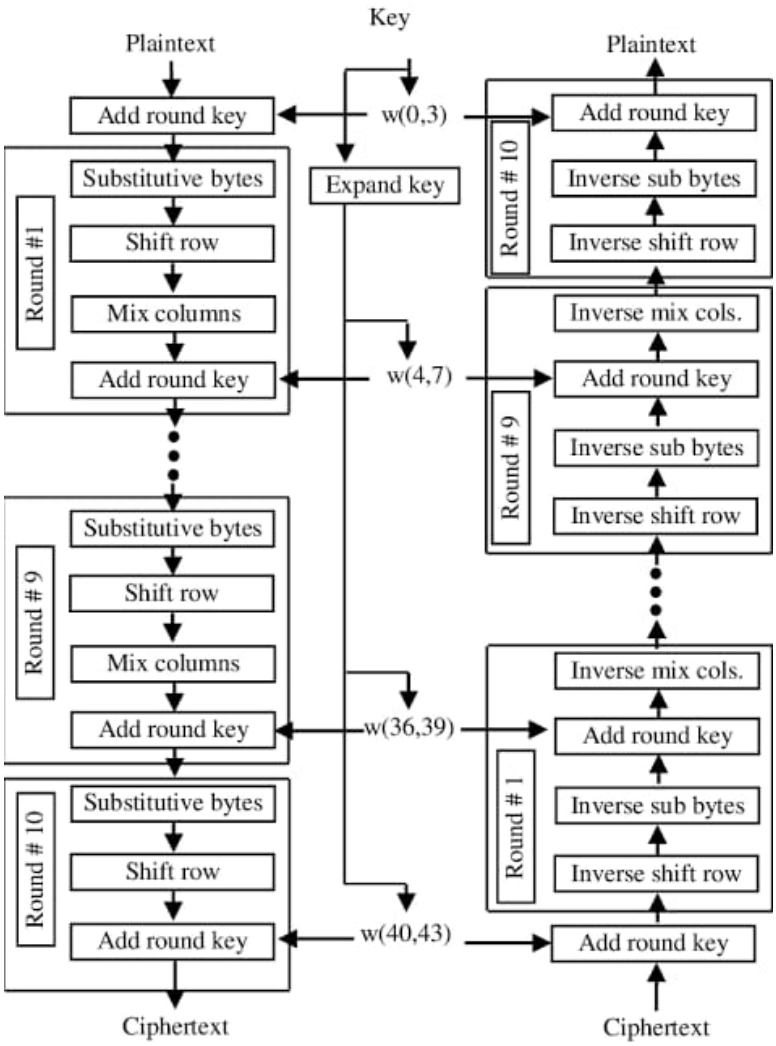
AES is based on a design principle known as a substitution–permutation network, and is efficient in both software and hardware. Unlike its predecessor DES, AES does not use a Feistel network. AES is a variant of Rijndael, with a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, Rijndael per se is specified with block and key sizes that may be any multiple of 32 bits, with a minimum of 128 and a maximum of 256 bits. The key size used for an AES cipher specifies the number of transformation rounds that convert the input, called the plaintext, into the final output,

called the ciphertext. Each round consists of several processing steps, including one that depends on the encryption key itself.

1.2 Implementation Environment

Careful choice must be made in selecting the mode of operation of the cipher. The simplest mode encrypts and decrypts each 128-bit block separately. In this mode, called electronic code book (ECB), blocks that are identical will be encrypted identically; this is entirely insecure. It makes some of the plaintext structure visible in the ciphertext. Selecting other modes, such as using a sequential counter over the block prior to encryption (i.e., CTR mode) and removing it after decryption avoids this problem. Another mode, Cipher Block Chaining (CBC) is one of the most commonly used modes of AES due to its use in TLS. CBC uses a random initialization vector (IV) to ensure that distinct ciphertexts are produced even when the same plaintext is encoded multiple times. The IV can be transmitted in the clear without jeopardizing security. A common practice is to prepend the 16 byte IV to the ciphertext, which gives the decrypter easy access to the IV. Care must be taken to use a new IV for every encryption operation, since otherwise an attacker can recover plaintext.

2 Procedure



Block diagram for AES-256 bits encryption and decryption

3 Major Components

JAVA code for Alice and Bob

```
    import java.util.Scanner;
import java.security.SecureRandom;
import java.util.Base64;
import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.IvParameterSpec;
```

```
import javax.crypto.spec.SecretKeySpec;
class Cryptography
{
public static void main(String[] args) throws Exception
{
KeyGenerator keyGenerator =
KeyGenerator.getInstance("AES");
keyGenerator.init(256);
// Generate Key
SecretKey key = keyGenerator.generateKey();
// Generating IV.
byte[] IV = new byte[16];
SecureRandom random = new SecureRandom();
random.nextBytes(IV);
Scanner in=new Scanner(System.in);
System.out.println("Enter the plainText to Encrypt");
String plainText=in.nextLine();
System.out.println("Original Text : "+plainText);
byte[] cipherText =
encrypt(plainText.getBytes(),key, IV);
System.out.println("Encrypted Text :
"+Base64.getEncoder().encodeToString(cipherText) );
String decryptedText =
decrypt(cipherText,key, IV);
System.out.println("DeCrypted Text : "+decryptedText);
}
public static byte[] encrypt (byte[] plaintext,SecretKey
key,byte[] IV ) throws Exception
{
//Get Cipher Instance
Cipher cipher=
Cipher.getInstance("AES/CBC/PKCS5Padding");
```

```
//Create SecretKeySpec
SecretKeySpec keySpec =
new SecretKeySpec(key.getEncoded(), "AES");
//Create IvParameterSpec
IvParameterSpec ivSpec = new IvParameterSpec(IV);
//Initialize Cipher for ENCRYPT_MODE
cipher.init(Cipher.ENCRYPT_MODE, keySpec, ivSpec);
//Perform Encryption
byte[] cipherText = cipher.doFinal(plaintext);
return cipherText;
}
public static String decrypt (byte[] cipherText, SecretKey
key,byte[] IV) throws Exception
{
//Get Cipher Instance
Cipher cipher =
Cipher.getInstance("AES/CBC/PKCS5Padding");
//Create SecretKeySpec
SecretKeySpec keySpec =
new SecretKeySpec(key.getEncoded(), "AES");
//Create IvParameterSpec
IvParameterSpec ivSpec = new IvParameterSpec(IV);
//Initialize Cipher for DECRYPT_MODE
cipher.init(Cipher.DECRYPT_MODE, keySpec, ivSpec);
//this Perform Decryption
byte[] decryptedText = cipher.doFinal(cipherText);
return new String(decryptedText);
}
}
```

4 Results

Output depicted in figure 1 and 2.

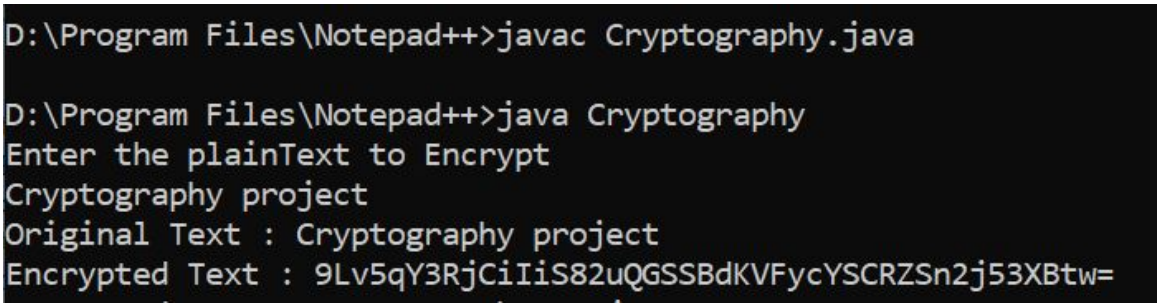


Figure 1: Computations at Sender-side (Alice).

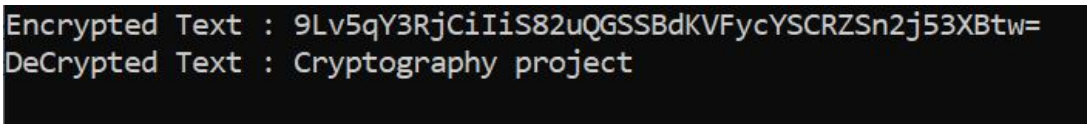


Figure 2: Computations at Receiver-side (Bob).

References

[1] Behrouz A Forouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", Mc Graw Hill, Third Edition, 2015.

[2] William Stallings, "Cryptography and Network Security: Principles and Practice", Pearson Education, Seventh Edition, 2017.

[3] Overleaf: a collaborative cloud-based LaTeX editor used for writing, editing and publishing scientific documents. `http://www.overleaf.com`.