javapapers

Java Symmetric AES Encryption Decryption using JCE

In this tutorial we will learn about AES symmetric encryption decryption using Java Cryptography Extension (JCE). In the previous tutorial we saw about encryption decryption using DES symmetric key algorithm. “Data Encryption Standard (DES)” is prone to brute-force attacks. It is a old way of encrypting data. It is replaced by “Advanced Encryption Standard (AES)”.

Main issue with DES was the short encryption key size. Shorter the key, it is easier to break it with brute force attack. AES can use 128, 192 or 256 bit encryption. The block size used by DES is 64 bits and by AES is 128 bits. Third difference is AES uses permutation substitution over the Feistel network used by DES.

If you are looking to choose between DES or AES for your real time application, AES is the way to go.

Example AES Symmetric Key Encryption Decryption

In comparison from the previous tutorial there are only two changes in the example program. One is the AES encryption and another is Base64 encoding.

If it were before Java 8, I would have used the Apache commons-code bundle for

Base64 encoding. In Java 8 we have got new classes in `java.util` package for Base64 encoding and decoding. It is important to encode the binary data with Base64 to ensure it to be intact without modification when it is stored or transferred.

```
        SecretKey secretKey = keyGenerator.generateKey();
        cipher = Cipher.getInstance("AES");

        String plainText = "AES Symmetric Encryption Decryption";
        System.out.println("Plain Text Before Encryption: " + plain

        String encryptedText = encrypt(plainText, secretKey);
        System.out.println("Encrypted Text After Encryption: " + en

        String decryptedText = decrypt(encryptedText, secretKey);
        System.out.println("Decrypted Text After Decryption: " + de
    }

    public static String encrypt(String plainText, SecretKey secret
            throws Exception {
        byte[] plainTextByte = plainText.getBytes();
        cipher.init(Cipher.ENCRYPT_MODE, secretKey);
        byte[] encryptedByte = cipher.doFinal(plainTextByte);
        Base64.Encoder encoder = Base64.getEncoder();
        String encryptedText = encoder.encodeToString(encryptedByte
        return encryptedText;
    }
```

Example Program Output

```
Plain Text Before Encryption: AES Symmetric Encryption Decryption
Encrypted Text After Encryption: sY6vkQrWRg0fvRzbqSAYxepeBIXg4AySj7X
Decrypted Text After Decryption: AES Symmetric Encryption Decryption
```

This Java tutorial was added on 02/11/2014.

va Symmetric Encryption Decryption using Java Cryptography Extension (JCE)

Simple Encryption Decryption with Modulo 26 Polyalphabetic Cipher

Comments on "Java Symmetric AES Encryption Decryption using JCE" Tutorial:

Java Symmetric Encryption Decryption using Java Cryptography Extension (JCE) says:

02/11/2014 at 10:24 pm

[…] is the latest encryption standard over the DES. You can refer the encryption decryption with AES symmetric algorithm using JCE tutorial. All the above given steps and concept are same, we just replace the DES with […]

Java File Encryption Decryption using Password Based Encryption (PBE) says: *10/11/2014 at 12:27 am*

[…] This tutorial is a continuation of our Java security series. In the previous tutorial we saw about symmetric AES encryption and decryption using Java Cryptography Extension (JCE). […]

hassan shahzad aheer says: *04/04/2015 at 12:42 am*

when i change keyGenerator.init(128); to keyGenerator.init(192); or keyGenerator.init(256);

it throws exception

Exception in thread “main” java.security.InvalidKeyException: Illegal key size or default parameters
at javax.crypto.Cipher.checkCryptoPerm(Cipher.java:1021)
at javax.crypto.Cipher.implInit(Cipher.java:796)
at javax.crypto.Cipher.chooseProvider(Cipher.java:859)
at javax.crypto.Cipher.init(Cipher.java:1229)
at javax.crypto.Cipher.init(Cipher.java:1166)
at algotesting.AlgoTesting.encrypt(AlgoTesting.java:52)

at algotesting.AlgoTesting.main(AlgoTesting.java:37)

---

ain says: *09/08/2015 at 4:41 pm*

i used netbean ide7, but theres an arror here Base64.Encoder encoder = Base64.getEncoder();

would you help me?thank you

Comments are closed for this "Java Symmetric AES Encryption Decryption using JCE" tutorial.

↑ Go to Top Site Map © 2008 - 2015 Java Papers