

java使用AES加密解密

原创辰辰呐 最后发布于2018-03-19 14:56:39 阅读数 343 收藏

展开

原文出自：<https://www.cnblogs.com/cheng-lhx/p/5817161.html>

```
import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;

import org.apache.commons.codec.binary.Base64;
/**
 *
 * @author Administrator
 *
 */
public class AES {

    // 加密
    public static String Encrypt(String sSrc, String sKey) throws Exception {
        if (sKey == null) {
            System.out.print("Key为空null");
            return null;
        }
        // 判断Key是否为16位
        if (sKey.length() != 16) {
            System.out.print("Key长度不是16位");
            return null;
        }
        byte[] raw = sKey.getBytes("utf-8");
        SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");//"算法/模式/补码方式"
        cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
        byte[] encrypted = cipher.doFinal(sSrc.getBytes("utf-8"));

        return new Base64().encodeToString(encrypted);//此处使用BASE64做转码功能，同时能起到2次加密的作用。
    }

    // 解密
```

```

public static String Decrypt(String sSrc, String sKey) throws Exception {
    try {
        // 判断Key是否正确
        if (sKey == null) {
            System.out.print("Key为空null");
            return null;
        }
        // 判断Key是否为16位
        if (sKey.length() != 16) {
            System.out.print("Key长度不是16位");
            return null;
        }
        byte[] raw = sKey.getBytes("utf-8");
        SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
        cipher.init(Cipher.DECRYPT_MODE, skeySpec);
        byte[] encrypted1 = new Base64().decode(sSrc);//先用base64解密
        try {
            byte[] original = cipher.doFinal(encrypted1);
            String originalString = new String(original,"utf-8");
            return originalString;
        } catch (Exception e) {
            System.out.println(e.toString());
            return null;
        }
    } catch (Exception ex) {
        System.out.println(ex.toString());
        return null;
    }
}

```

```

public static void main(String[] args) throws Exception {
    /*
     * 此处使用AES-128-ECB加密模式， key需要为16位。
     */
    String cKey = "1234567890123456";
    // 需要加密的字串
    String cSrc = "www.gowhere.so";
    System.out.println(cSrc);
    // 加密
    String enString = AES.Encrypt(cSrc, cKey);
}

```

```
System.out.println("加密后的字串是：" + enString);

// 解密
String DeString = AES.Decrypt(enString, cKey);
System.out.println("解密后的字串是：" + DeString);
    }
}
```

//源代码片段来自云代码<http://yuncode.net>

---

版权声明：本文为CSDN博主「辰辰呐」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载  
请附上原文出处链接及本声明。

原文链接：[https://blog.csdn.net/qq\\_37581708/article/details/79611729](https://blog.csdn.net/qq_37581708/article/details/79611729)