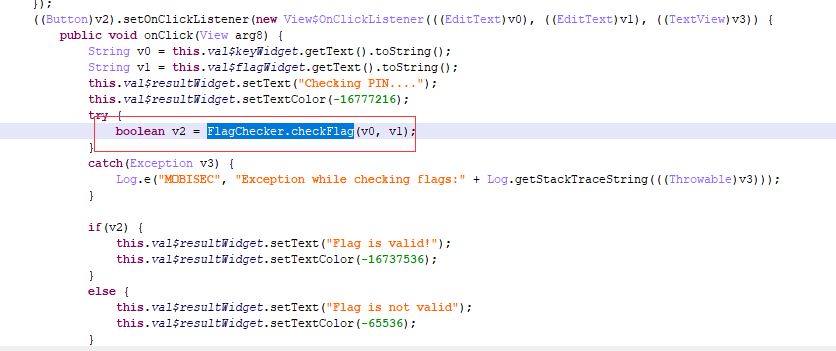
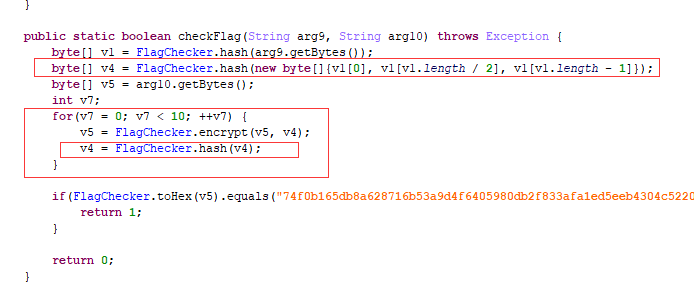
# 题目分析

拖进jeb进行分析，app逻辑很清晰，让输入key和flag。FlagChecker进行验证。



分析FlagChecker，发现进行了十次aes加密，第n次aes加密的key就是对输入key的三个字节进行第n次hash加密的结果。

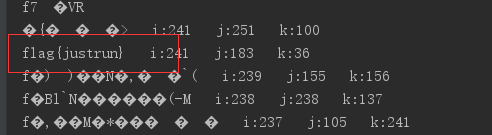


因为key就三个字节，并且知道flag格式位flag{xxxxxxx}我们进行爆破，从前面和从后面同时爆破，发现从后面很快出flag了。

# 脚本

放出从后往前爆破的脚本。

import javax.crypto.Cipher;  
import javax.crypto.CipherOutputStream;  
import javax.crypto.spec.SecretKeySpec;  
import java.io.ByteArrayOutputStream;  
import java.security.Key;  
import java.security.MessageDigest;  
  
public class Demo1{  
 public static void main(String[] args) throws Exception {  
 String encrypted = "74f0b165db8a628716b53a9d4f6405980db2f833afa1ed5eeb4304c5220bdc0b541f857a7348074b2a7775d691e71b490402621e8a53bad4cf7ad4fcc15f20a8066e087fc1b2ffb21c27463b5737e34738a6244e1630d8fa1bf4f38b7e71d707425c8225f240f4bd2b03d6c2471e900b75154eb6f9dfbdf5a4eca9de5163f9b3ee82959f166924e8ad5f1d744c51416a1db89638bb4d1411aa1b1307d88c1fb5";  
  
 for(char i = 255;i > 1;i--)  
 {  
 for(char j = 255;j > 0;j--)  
 {  
 for(char k = 255;k > 0;k--)  
 {  
 byte[] bkey1 = new byte[]{(byte)i,(byte)j,(byte)k};  
 byte[] bflag = *fromHextobytes*(encrypted);  
 for(int l = 10; l > 0 ;l--)  
 {  
 byte[] bkey2 = *hash*(bkey1);  
 for(int m = 0; m < l-1; m++)  
 {  
 bkey2 = *hash*(bkey2);  
 }  
 bflag = *decrypt*(bflag, bkey2);  
 }  
 if(new String(bflag,"utf-8").indexOf("f") == 0)//过滤结果以f开头的  
 System.*out*.println(new String(bflag,"utf-8") + " i:" + (int)i + " j:" + (int)j + " k:" + (int)k );  
 }  
 }  
 }  
 }  
 public static byte[] decrypt(byte[] in, byte[] key) throws Exception {  
 Key aesKey = new SecretKeySpec(key, "AES");  
 Cipher decryptCipher = Cipher.*getInstance*("AES/ECB/PKCS5Padding");  
 decryptCipher.init(Cipher.*DECRYPT\_MODE*, aesKey);  
 ByteArrayOutputStream outputStream = new ByteArrayOutputStream();  
 CipherOutputStream cipherOutputStream = new CipherOutputStream(outputStream, decryptCipher);  
 cipherOutputStream.write(in);  
 cipherOutputStream.flush();  
 cipherOutputStream.close();  
 return outputStream.toByteArray();  
 }  
 public static byte[] hash(byte[] in) throws Exception {  
 MessageDigest md = MessageDigest.*getInstance*("MD5");  
 md.update(in);  
 return md.digest();  
 }  
  
 public static byte[] fromHextobytes(String hex) {  
 hex = hex.length() % 2 != 0 ? "0" + hex : hex;  
  
 byte[] b = new byte[hex.length() / 2];  
 for (int i = 0; i < b.length; i++) {  
 int index = i \* 2;  
 int v = Integer.*parseInt*(hex.substring(index, index + 2), 16);  
 b[i] = (byte) v;  
 }  
 return b;  
 }  
}



Flag：flag{justrun}