【逆向学习笔记3】Android逆向刷题（攻防世界新手区）持更

原创

:)) 于 2020-08-08 14:18:16 发布 230 收藏

本文链接：https://blog.csdn.net/weixin_45927195/article/details/107788787

版权

攻防世界新手区

函数解析

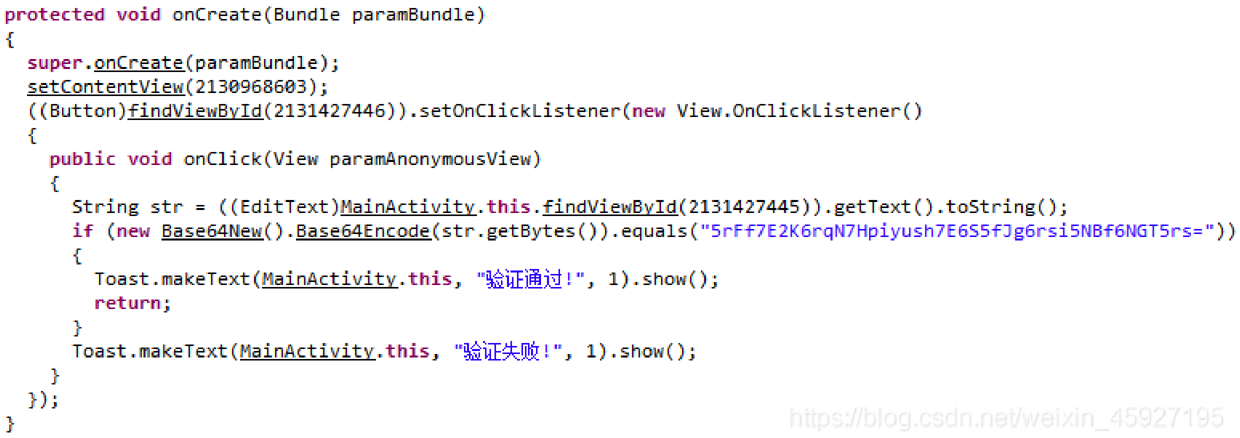
setContentView(2130978703)

findViewById(2131427415)

题目writeup

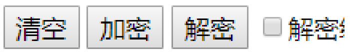
easy-apk(根据指定码表写Base64解密脚本)

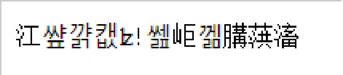
首先用逆向助手将dex文件转成jar文件后，在JD-GUI中打开，查看主活动：



```
protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130968603);
  ((Button)findViewById(2131427446)).setOnClickListener(new View.OnClickListener()
  {
    public void onClick(View paramAnonymousView)
    {
      String str = ((EditText)MainActivity.this.findViewById(2131427445)).getText().toString();
      if (new Base64New().Base64Encode(str.getBytes()).equals("5rFf7E2K6rqN7Hpiyush7E6S5fJg6rsi5NBf6NGT5rs="))
      {
        Toast.makeText(MainActivity.this, "验证通过!", 1).show();
        return;
      }
      Toast.makeText(MainActivity.this, "验证失败!", 1).show();
    }
  });
}
```

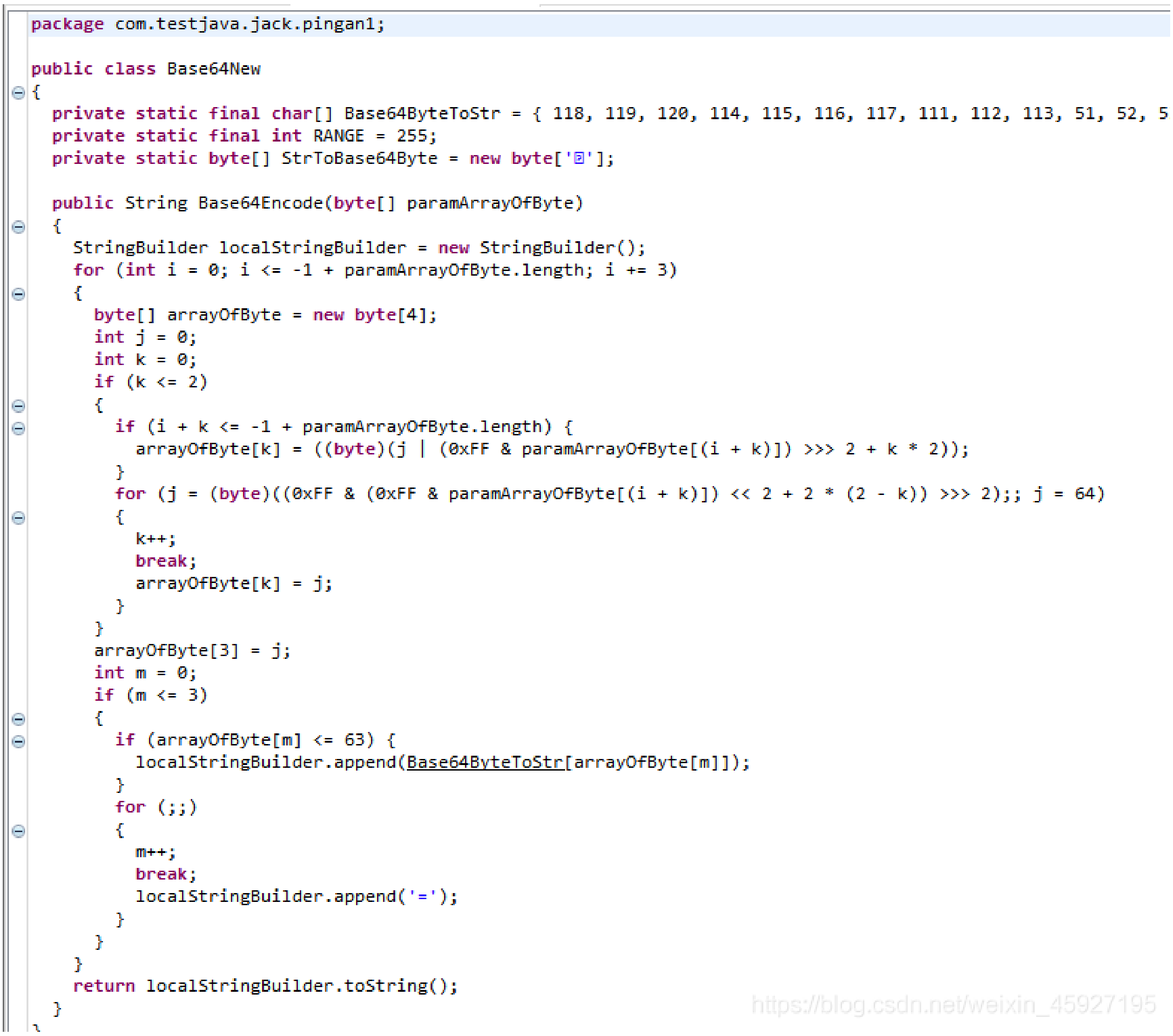
可以看到`5rFf7E2K6rqN7Hpiyush7E6S5fJg6rsi5NBf6NGT5rs=`是经过Base64加密得到的，但是直接拿去解密会得到乱码：





江쌀갉갮ㅌ! 쐢岠겗膳蒩滀

再仔细看这个if条件，会发现该类的名字多了个New，猜测是用的是不同平常的Base64加密过程，点进去查看：



```
package com.testjava.jack.pingan1;

public class Base64New
{
  private static final char[] Base64ByteToStr = { 118, 119, 120, 114, 115, 116, 117, 111, 112, 113, 51, 52, 5
  private static final int RANGE = 255;
  private static byte[] StrToBase64Byte = new byte['?'];

  public String Base64Encode(byte[] paramArrayOfByte)
  {
    StringBuilder localStringBuilder = new StringBuilder();
    for (int i = 0; i <= -1 + paramArrayOfByte.length; i += 3)
    {
      byte[] arrayOfByte = new byte[4];
      int j = 0;
      int k = 0;
      if (k <= 2)
      {
        if (i + k <= -1 + paramArrayOfByte.length) {
          arrayOfByte[k] = ((byte)(j | (0xFF & paramArrayOfByte[(i + k)]) >>> 2 + k * 2));
        }
        for (j = (byte)((0xFF & (0xFF & paramArrayOfByte[(i + k)]) << 2 + 2 * (2 - k)) >>> 2);; j = 64)
        {
          k++;
          break;
          arrayOfByte[k] = j;
        }
      }
      arrayOfByte[3] = j;
      int m = 0;
      if (m <= 3)
      {
        if (arrayOfByte[m] <= 63) {
          localStringBuilder.append(Base64ByteToStr[arrayOfByte[m]]);
        }
        for (;;)
        {
          m++;
          break;
          localStringBuilder.append('=');
        }
      }
    }
    return localStringBuilder.toString();
  }
```

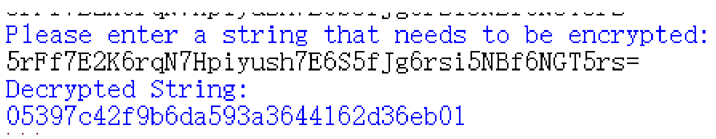
根据这个加密过程重新写一个Base64解密过程：
（参考了这篇Base64加解密过程的基本代码）

```
# coding:utf-8
s = "vwxrstuopq34567ABCDEFGHIJyz012PQRSTKLMNOZabcdUVWXYefghijklmn89+/"

def My_base64_decode(inputs):
 # 将字符串转化为2进制
 bin_str = []
 for i in inputs:
  if i != '=':
   x = str(bin(s.index(i))).replace('0b', '')
   bin_str.append('{:0>6}'.format(x))
 #print(bin_str)
 # 输出的字符串
 outputs = ""
 nums = inputs.count('=')
 while bin_str:
  temp_list = bin_str[:4]
  temp_str = "".join(temp_list)
  #print(temp_str)
  # 补足8位字节
  if(len(temp_str) % 8 != 0):
   temp_str = temp_str[0:-1 * nums * 2]
  # 将四个6字节的二进制转换为三个字符
  for i in range(0,int(len(temp_str) / 8)):
   outputs += chr(int(temp_str[i*8:(i+1)*8],2))
  bin_str = bin_str[4:]
 print("Decrypted String:\n%s "%outputs)

input_str = input("Please enter a string that needs to be encrypted: \n")
My_base64_decode(input_str)
```

运行，解得：



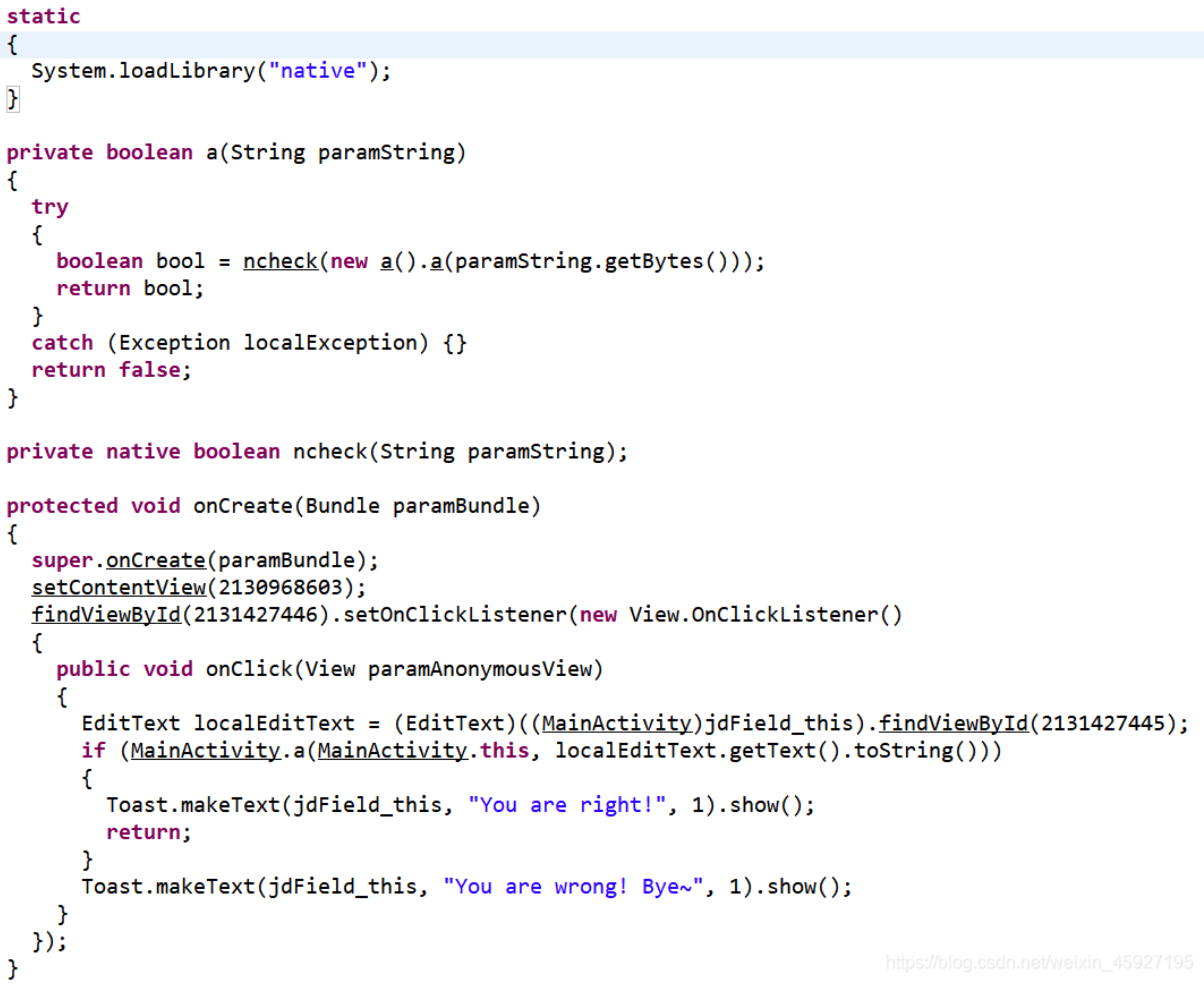
```
Please enter a string that needs to be encrypted:
5rFf7E2K6rqN7Hpiyush7E6S5fJg6rsi5NBf6NGT5rs=
Decrypted String:
05397c42f9b6da593a3644162d36eb01
```

答案就是`flag{05397c42f9b6da593a3644162d36eb01}`。

（附上标准Base64加密原理）

easyjni

首先用逆向助手将dex文件转成jar文件后，在JD-GUI中打开，查看主活动：



```java
static
{
  System.loadLibrary("native");
}

private boolean a(String paramString)
{
  try
  {
    boolean bool = ncheck(new a().a(paramString.getBytes()));
    return bool;
  }
  catch (Exception localException) {}
  return false;
}

private native boolean ncheck(String paramString);

protected void onCreate(Bundle paramBundle)
{
  super.onCreate(paramBundle);
  setContentView(2130968603);
  findViewById(2131427446).setOnClickListener(new View.OnClickListener()
  {
    public void onClick(View paramAnonymousView)
    {
      EditText localEditText = (EditText)((MainActivity)jdField_this).findViewById(2131427445);
      if (MainActivity.a(MainActivity.this, localEditText.getText().toString()))
      {
        Toast.makeText(jdField_this, "You are right!", 1).show();
        return;
      }
      Toast.makeText(jdField_this, "You are wrong! Bye~", 1).show();
    }
  });
}
```

(未完待续…)