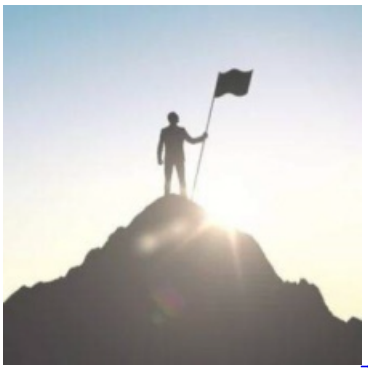
XCTF MOBILE 新手 easyjni

原创

A_dmins 于 2019-10-09 19:07:43 发布 426 收藏

分类专栏： CTF题 XCTF

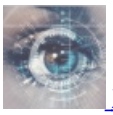
本文链接：https://blog.csdn.net/qq_42967398/article/details/102467762



CTF题 同时被 2 个专栏收录

115 篇文章 11 订阅

订阅专栏



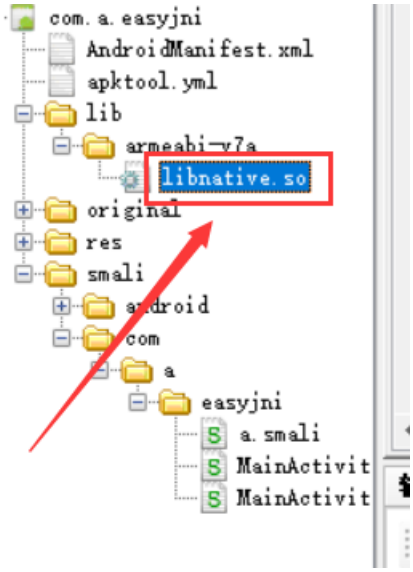
XCTF

24 篇文章 0 订阅

订阅专栏

XCTF MOBILE 新手 easyjni

下载文件，发现是个APK文件，直接APKIDE分析一波，发现有lib，看一看名字：



直接java反编译，找到主函数：

```
public class MainActivity
  extends c
{
  static
```

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

  private boolean a(String paramString)
  {
    try
    {
      boolean bool = ncheck(new a().a(paramString.getBytes()));
      return bool;
    }
    catch (Exception paramString) {}
    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)
      {
        paramAnonymousView = (EditText)((MainActivity)jdField_this).findViewById(2131427445);
        if (MainActivity.a(MainActivity.this, paramAnonymousView.getText().toString()))
        {
          Toast.makeText(jdField_this, "You are right!", 1).show();
          return;
        }
        Toast.makeText(jdField_this, "You are wrong! Bye~", 1).show();
      }
    });
  }
}
```


调用了lib里面的库。同时还调用了a类：

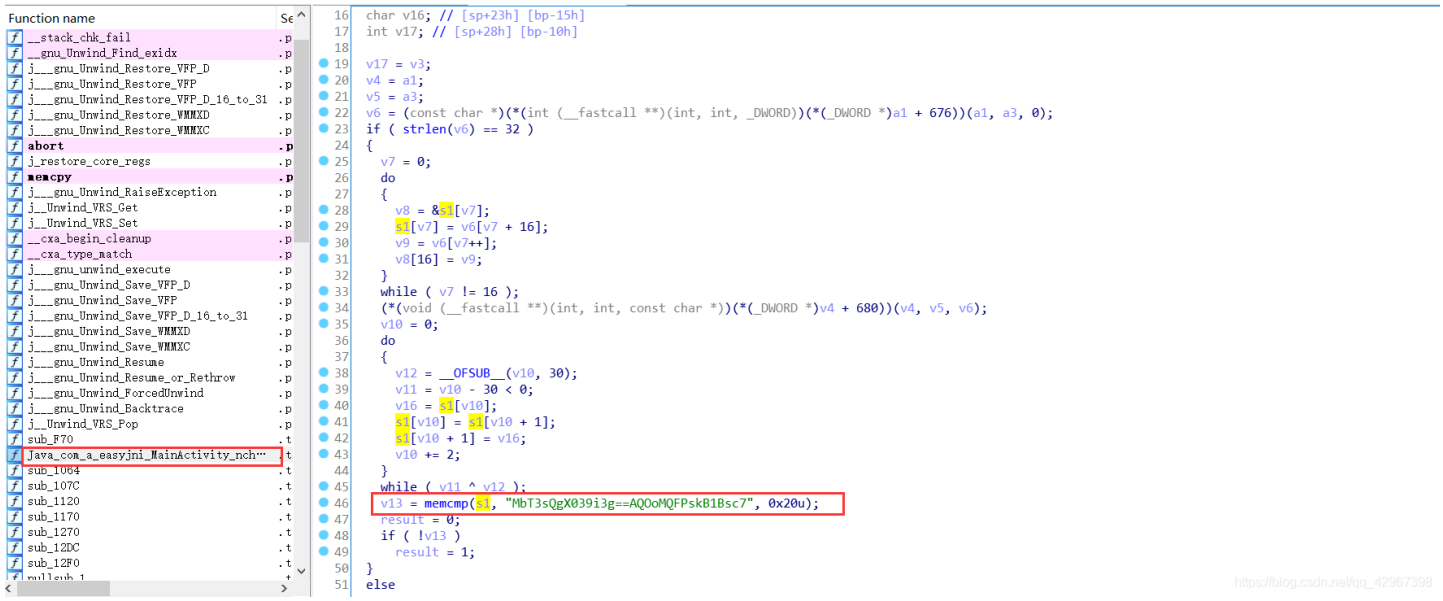
```
{
  private static final char[] a = { 105, 53, 106, 76, 87, 55, 83, 48, 71, 88, 54, 117, 102, 49, 99, 118, 51, 110, 121, 52, 113, 56, 101, 115, 50, 81, 43, 98, 100, 107, 8

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

母庸置疑，和我们之前做的一个base64变码表的一样，应该也是一个变换码表的base64：

```
>>> a = [105, 53, 106, 76, 87, 55, 83, 48, 71, 88, 54, 117, 102, 49, 99, 118, 51, 110, 121, 52, 113, 56, 101, 115, 50,
1, 43, 98, 100, 107, 89, 103, 75, 79, 73, 84, 47, 116, 65, 120, 85, 114, 70, 108, 86, 80, 122, 104, 109, 111, 119, 57,
6, 72, 67, 77, 68, 112, 69, 97, 74, 82, 90, 78 ]
>>> flag = ""
>>> for i in a:
...     flag += chr(i)
...
>>> flag
'i5jLW7S0GX6uf1cv3ny4q8es2Q+bdkYgKOIT/tAxUrF1VPzhmow9BHCMDpEaJRZN'
>>>
```

进入到libc中的native中看看，解压后用ida打开：



emmmm，好像逻辑还是挺简单的，前面16个和后面的16个换一下位置，然后两两交换一下位置：

```
>>> s = "AQOoMQFPskB1Bsc7MbT3sQgX039i3g=="
>>> x = ""
>>> k = 0
>>> for i in range(0,16):
...     k += 2
...     x += s[k-1]
...     x += s[k-2]
...
>>> x
'QAoOQMPFks1BsB7cbM3TQsXg30i9g3=='
>>>
```


得到一个字符串：`QAoOQMPFks1BsB7cbM3TQsXg30i9g3==`

使用上次自己写的base64变换码表脚本，变换一下码表，跑一下：

```
python .\base64.py

    ************************************
    *    (1)encode        (2)decode    *
    ************************************

Please select the operation you want to perform:
2
Please enter a string that needs to be decrypted:
QAoOQMPFks1BsB7cbM3TQsXg30i9g3==
Decrypted String:
```

get flag：`flag{just_ANot#er_@p3}`