

# 第56天：攻防世界-Mobile-Flag\_system

原创

Sllenc3 于 2019-12-25 23:45:09 发布 533 收藏

分类专栏: [Android](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_41858371/article/details/103707025](https://blog.csdn.net/qq_41858371/article/details/103707025)

版权



[Android 专栏收录该内容](#)

30 篇文章 1 订阅

订阅专栏

1.做了两道题, 一个没做出来。自闭了, 一个人学习, 时间全浪费了, 学习效率降低几倍。

flag system做了一半。

首先拿到一个文件, 没有后缀, 010editor查看,

(这是更改后的)

```
0123456789ABCDEF
ANDROID BACKUP.1
.1.none.xÚì¹.P`a
```

android备份文件, 之前见到过, 改后缀为.ab文件。

然后使用abe.jar, 但是发现没什么用, 直接看writeup, 知道是abe的问题。查看backupdecrypt.pl

```
! LINE. STRUCTURE OF THE ADB BACKUP HEADER          EXAMPLE
! 1. Magic:                ANDROID BACKUP
! 2. Version (1 only):     1
! 3. Compression (1=compressed): 1
! 4. Encryption (AES-256/none): AES-256
! -----
```

感觉就是version和compress的问题, 对应着其他的ab文件, 改header中的数据。

```
000h: 41 4E 44 52 4F 49 44 20 42 41 43 4B 55 50 0A 31  ANDROID BACKUP.1
010h: 0A 31 0A 6E 6F 6E 65 0A 78 DA EC B9 07 50 93 61  .1.none.xÚì¹.P`a
```

然后使用命令 `java -jar abe.jar unpack Flag_system.ab 1.tar`

将1.tar解压后会得到两个apk。

```
com.example.mybackup 201
com.example.zi 201
```

zi中的apk没发现flag。

分析mybackup。

```

  com
  └─ example
     └─ mybackup
        ├── BooksDB
        ├── BuildConfig
        ├── R
        ├── SQLiteDatabaseDemo
        └─ Test
    ├── google
    └─ example
       ├── EventDataSQLHelper
       └─ SQLDemoActivity
    ├── net
    └─ org
       └─ apache

```

sqlcipher加密数据库。

```

    if (this.mCursor != null) {
        this.mCursor.moveToPosition(arg6);
        this.BOOK_ID = this.mCursor.getInt(0);
        if ("Flag".equals(this.mCursor.getString(1))) {
            this.BookName.setText("Guess");
            this.BookAuthor.setText("Flag is here!");
        }
        else {
            this.BookName.setText(this.mCursor.getString(1));
            this.BookAuthor.setText(this.mCursor.getString(2));
        }
    }
}

```

```

    ic boolean onOptionsItemSelected(MenuItem arg2) {

```

猜测flag就在这BOOKS.db中，所以需要拿到key。

```

    this.k = Test.getSign(arg4);
    this.db = this.getWritableDatabase(this.k);
    this.dbr = this.getReadableDatabase(this.k);

```

分析代码，key就是取apk的签名信息

```

public static String getSign(Context arg7) {
    Object v3;
    Iterator v1 = arg7.getPackageManager().getInstalledPackages(0x40).iterator();
    do {
        if (v1.hasNext()) {
            v3 = v1.next();
            if (!((PackageInfo)v3).packageName.equals(arg7.getPackageName())) {
                continue;
            }
            break;
        }
        else {
            return "";
        }
    }
    while (true);
    String v5 = ((PackageInfo)v3).signatures[0].toCharsString();
    Log.i("Hello", v5);
    return Test.SHA1(v5);
}

```

因为不知道签名到底是什么，所以修改smali代码，用日志打印变量。

```
const-string v1, "Hello"  
  
invoke-static {v1, v5}, Landroid/util/Log;->i(Ljava/lang/String;Ljava/lang/String;)I
```

发现signature就是META-INF\CERT.RSA文件的16进制数据。

SHA-1加密后为：7087d05a7aee9efd3c7ad6636784d7b71b040b0a

至此，我们拿到了key。

但是还是打不开数据库，writeup说要是用正确版本，找了几个小时的版本，我就纳闷了，writeup提一下不行吗。。。

最终还是放弃了，我是个辣鸡，唉。。。。

求大佬们教教怎么做吧。