ctrip tech 携程技术中心

ctrip tech 携程技术中心 | IT大咖说 知识分享平台

携程技术沙龙

移动安全自动化审计之路

分享人：谢鑫

讲师介绍

谢鑫

- 北京墨云科技有限公司首席安全官、联合创始人
  - 产品架构设计
  - 大数据安全分析
  - 安全体系建设
- 百度安全实验室安全专家
  - 黑产打击
  - 自动化漏洞挖掘
  - 漏洞研究
  - 威胁情报建设

目录

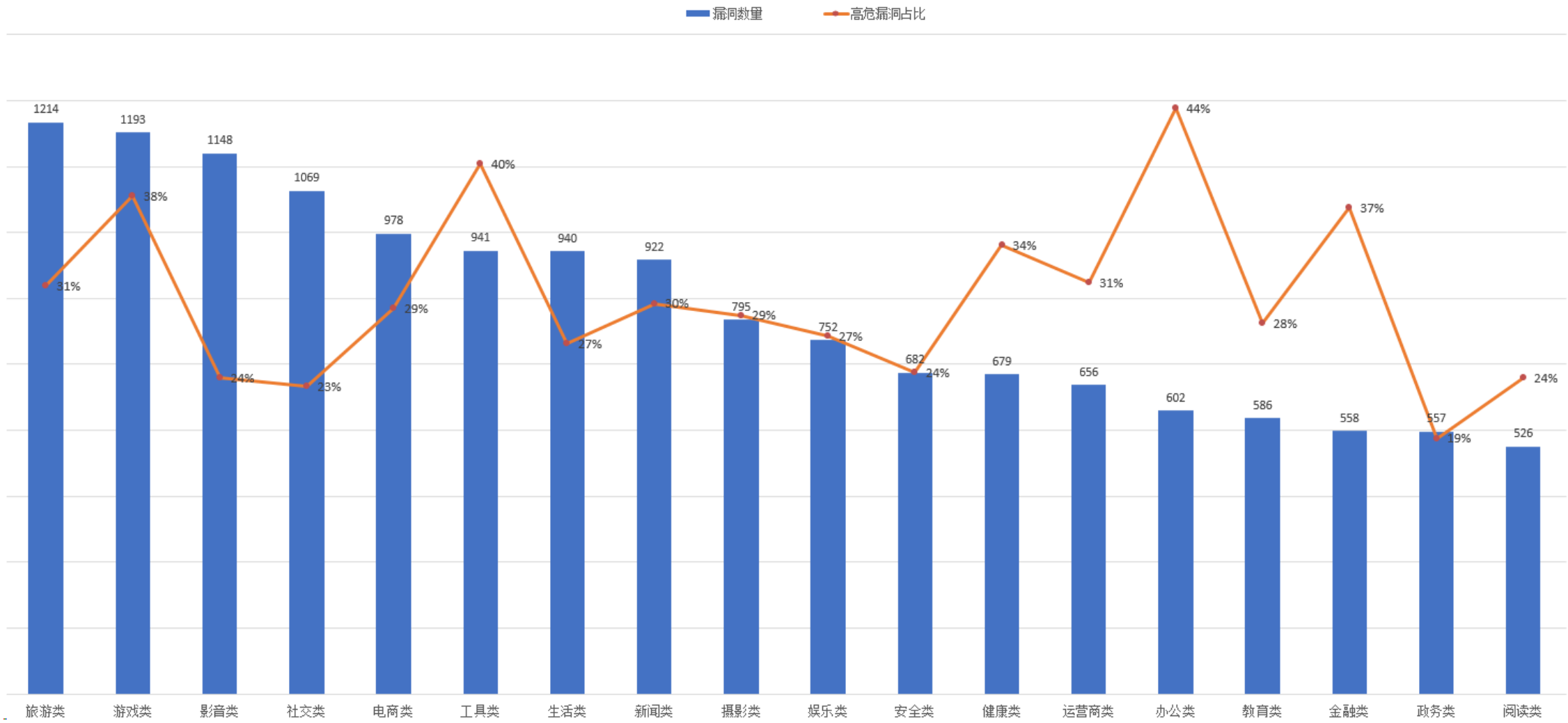
CONTENTS

遇到过移动安全风险造成的个人隐私泄露？

安全漏洞会对企业造成什么影响？

移动端黑色产业链如何盈利？

2016年18个类别Top10应用漏洞数量图



Android 应用漏洞风险

开发者窃取用户隐私犹如“探囊取物”，数据库中所有信息都会被开发者获取，包括用户的邮箱信息、QQ 和 MSN 等手机程序的信息，甚至用户喜欢哪种类别的应用程序等。而这些很可能成为信息买卖产业链的源头。

黑色产业

灰色产业

短信劫持

静默下载

静默流量

恶意扣费

数据窃取

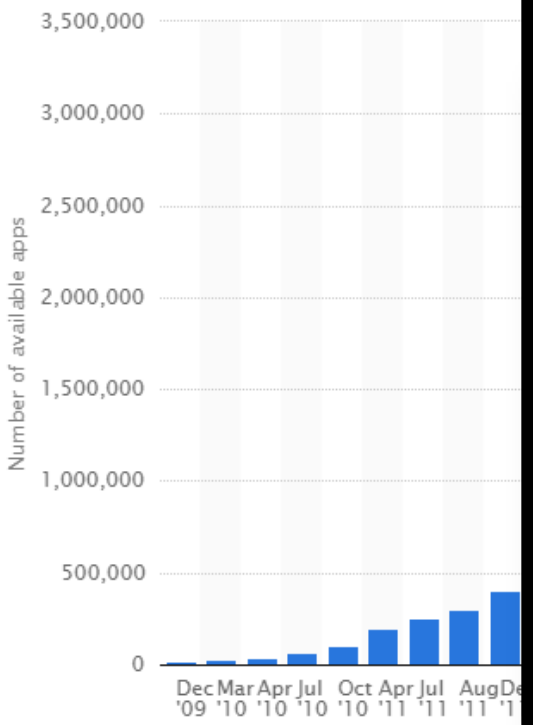
羊毛党

黄牛党

隐私获取

二次打包

二次打包产业链

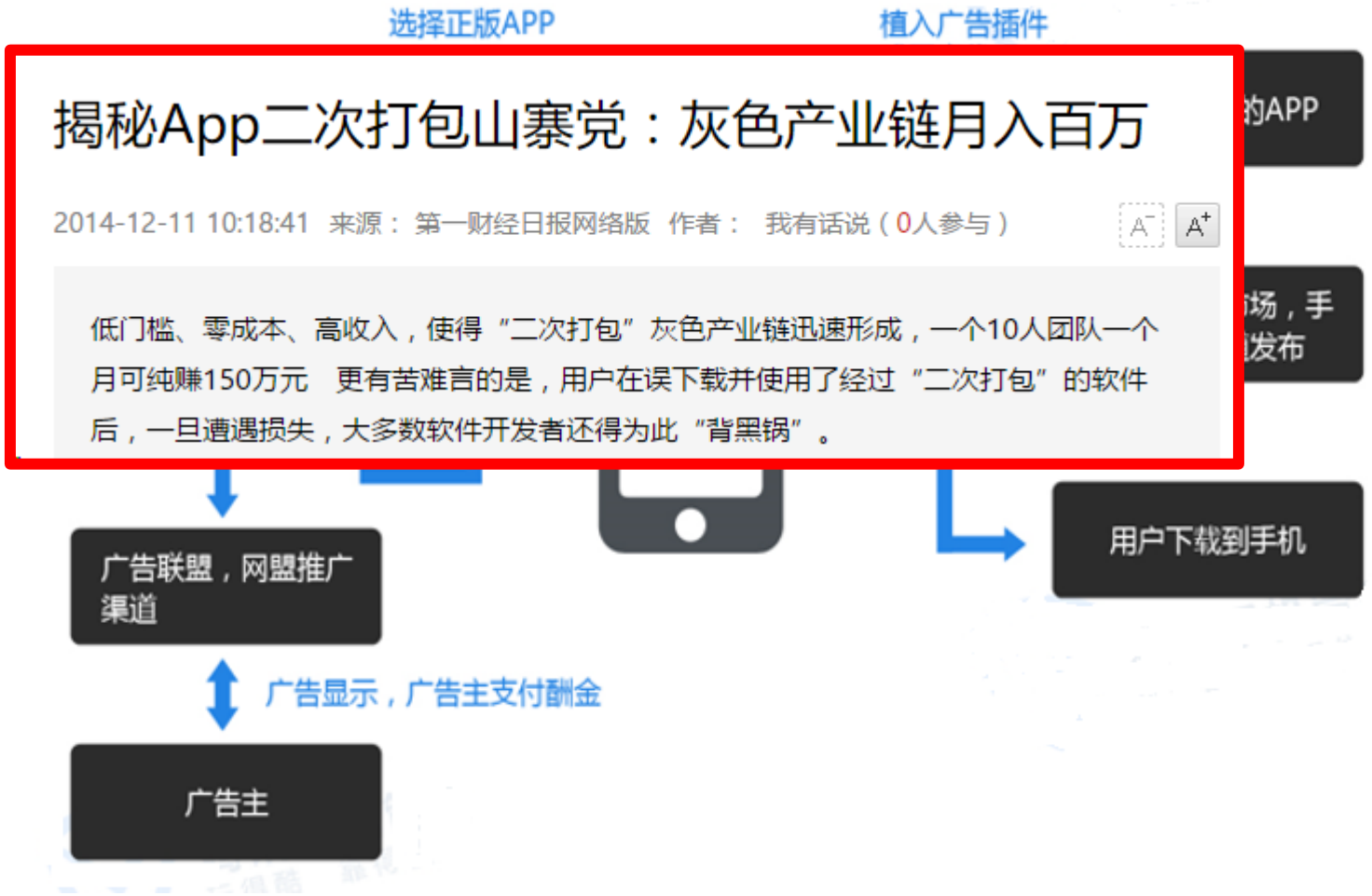


Google应用商店09年12月

在仿冒

- 平均每个应用仿冒量达54个
- 59%具有恶意行为

二次打包背后操作利益链



我们需要了解

- 需要什么工具进行检测
- 如何使用这些工具
- 通过什么方式检测是否存在漏洞
- 了解漏洞风险成因
- 了解漏洞风险的威胁级别

Apk文件结构

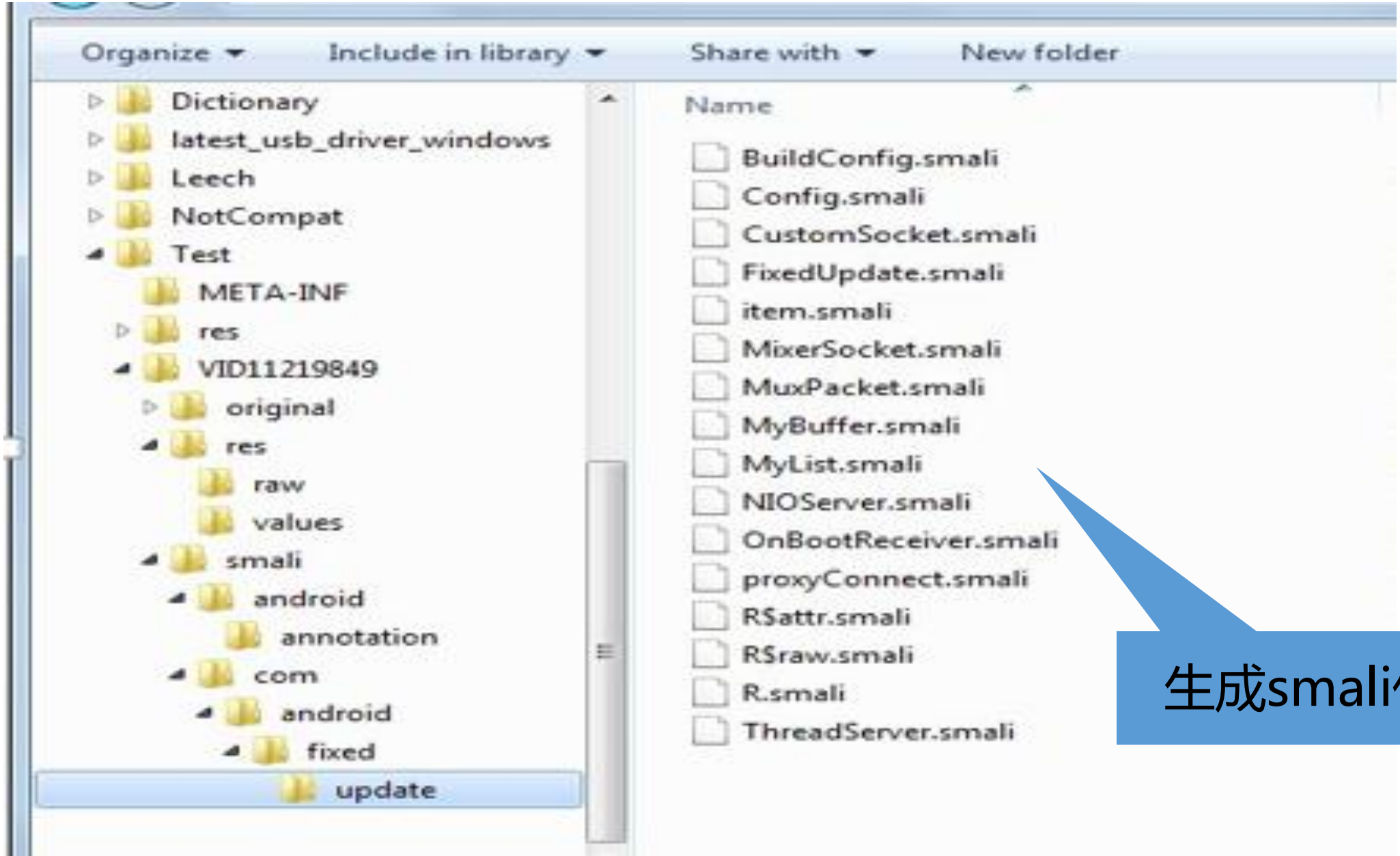
- APK是一种基于ZIP的文件格式
- 主要组成部分
  - 应用配置文件 AndroidManifest.xml
  - Dalvik字节码执行文件 classes.dex
  - 资源配置文件 Resources.arsc
  - 程序依赖库 Libs/
  - 应用签名、证书 META-INF/

静态分析工具

- Apktool　apk反编译工具
- Dex2jar　dex反编译jar工具
- Jd-gui　jar反编译java工具
- ZjDroid　脱壳工具

```
apktool d whatsapp.apk
I: Using Apktool 2.0.0-RC3 on whatsapp.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Users\As
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

反编译apk文件


Organize
Include in library
Share with
New folder
Dictionary
latest_usb_driver_windows
Leech
NotCompat
Test
META-INF
res
VID11219849
original
res
raw
values
smali
android
annotation
com
android
fixed
update
Name
BuildConfig.smali
Config.smali
CustomSocket.smali
FixedUpdate.smali
item.smali
MixerSocket.smali
MuxPacket.smali
MyBuffer.smali
MyList.smali
NIOServer.smali
OnBootReceiver.smali
proxyConnect.smali
RSattr.smali
RSraw.smali
R.smali
ThreadServer.smali
生成smali代码

C:\Users\kevinkm\Desktop\Test\VID11219849\smali\com\android\fixed\update\MixerSocket.smali - Notepad2

File Edit View Tools Help

```
    :sswitch_1
    iget v3, v1, Lcom/android/fixed/update/MuxPacket;->chanal:I

    invoke-virtual {p0, v3}, Lcom/android/fixed/update/MixerSocket;->shutdowChanal(I)V

    goto :goto_6

    .line 491
    :sswitch_2
    invoke-virtual {p0}, Lcom/android/fixed/update/MixerSocket;->se     ong()V

    goto :goto_6

    .line 495
    :sswitch_3
    iget-object v3, v1,
Lcom/android/fixed/update/MuxPacket;->Data:Lcom/android/fixed/update/MyBuffer;

    invoke-virtual {v3}, Lcom/android/fixed/update/MyBuffer;->array()[B

    move-result-object v3

    invoke-virtual {p0, v3}, Lcom/android/fixed/update/MixerSocket;->setTimeOut([B)V

    goto :goto_6

    .line 498
```

Ln 1 : 1,194 Col 1 Sel 0 | 29.32 KB | ANSI | CR+LF | INS | Default Text

smali代码

APP安全检测内容（部分）

应用安全

防调试
防重打包
组件访问权限
组件安全漏洞
恶意行为检测

用户操作安全

弱口令检测
验证码安全检测
登录限制检测
密码保护检测

通信安全

传输协议分析
身份认证检测
断网会话检测
重放攻击检测

数据安全

信息显示检测
本地存储安全检测
键盘劫持检测
防截屏录像检测

服务器安全

SQL注入检测
跨站攻击检测
目录遍历检测
后门检测
服务器端口开放漏洞

环境安全

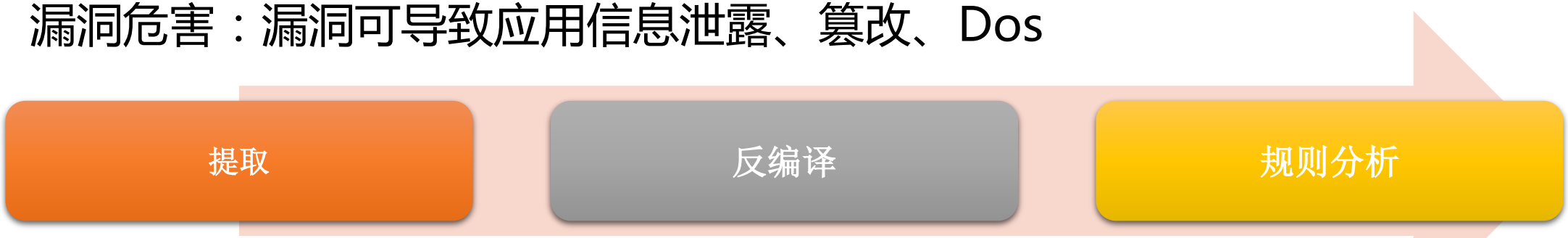
网络环境安全检测

业务安全

越权访问
信息提示检测
短信炸弹攻击

基于smali代码分析

漏洞名称：Shared Preferences任意读写漏洞
漏洞成因：错误设置getSharedPreferences函数mode参数为非0值
漏洞危害：漏洞可导致应用信息泄露、篡改、Dos




```
:goto_0
return-void
:cond_0
neg-int v0, v10
iput v0, v9, Ld/a/b;->x:I
const/4 v2, 0x3
invoke-virtual {p0, v1, v2}, Landroid/content/Context;->getSharedPreferences(Ljava/lang/String;I)Landroid/content/SharedPreferences;
```

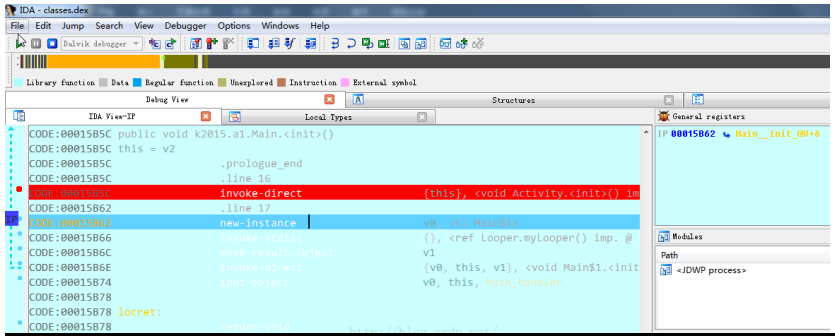
2. 分析寄存器赋值，判断是否为0x1，0x2，0x3

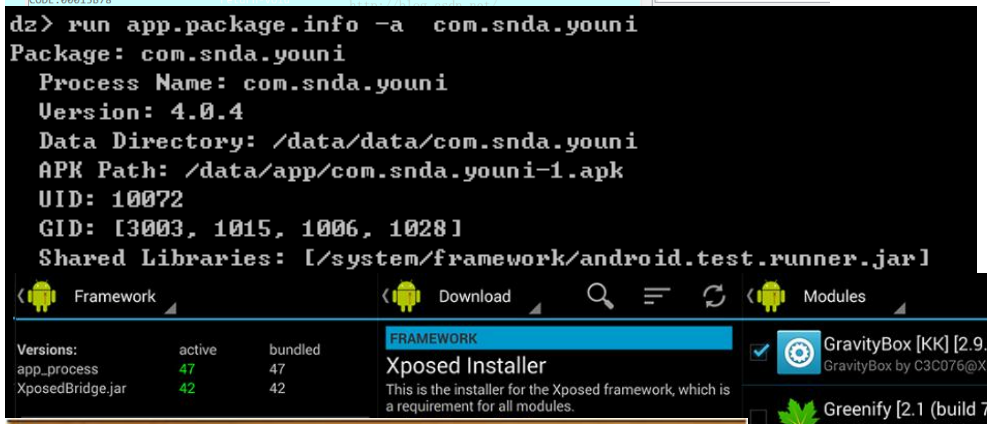
1. 关键函数

1. 识别精度高，漏报少
2. 减少反编译步骤，提高效率
3. 编写漏洞指纹相对简单
4. 快速定位到漏洞位置

动态分析工具

ADB	Android debug工具
IDA	反汇编软件
Drozer	安全测试框架
intentFuzzer	Intent fuzz测试工具
Xposed	Android Hook框架





动态分析方法

应用行为监控测试

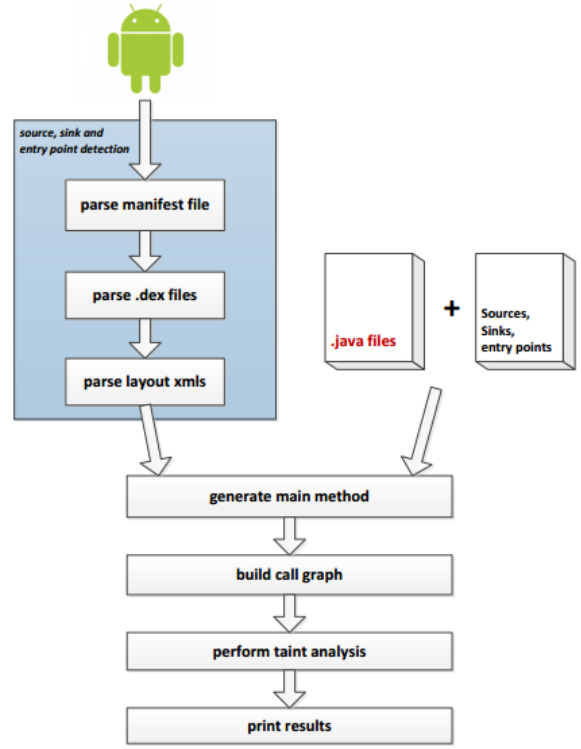
- 监控 Android应用实时运行行为
- 分析Android应用交互行为风险

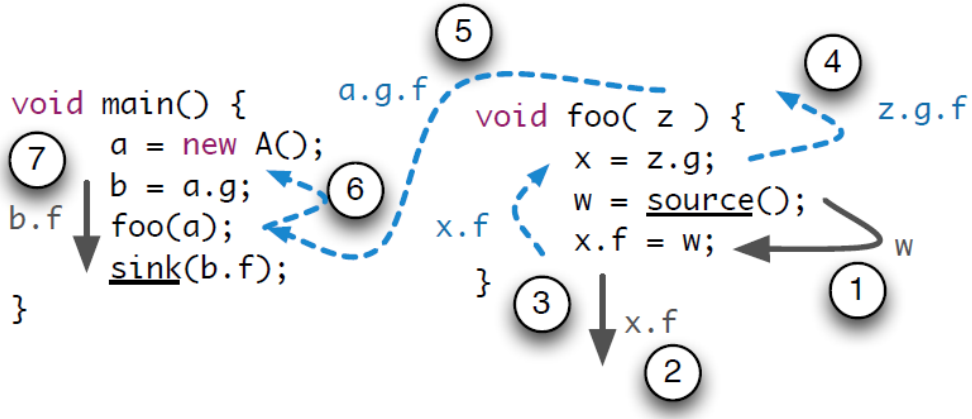
模糊Fuzz测试

- 挖掘系统组件漏洞
- DOS拒绝服务漏洞

污点分析测试

- 跟踪污点数据的传播过程
- 漏洞是否在实际环境中是否存在





静态分析优缺点

优点：

- 全代码覆盖
- 分析速度快、轻量级
- 无需依赖执行环境
- 漏洞覆盖面广

缺点：

- 误报率高
- 对于加壳应用，需要逆向脱壳
- 交互类，数据传输类漏洞无法测试

动态分析优缺点

优点：

- 动态监控应用行为
- 精确安全分析，误报率低
- 无视应用加壳
- 可对交互类，数据传输类漏洞进行测试

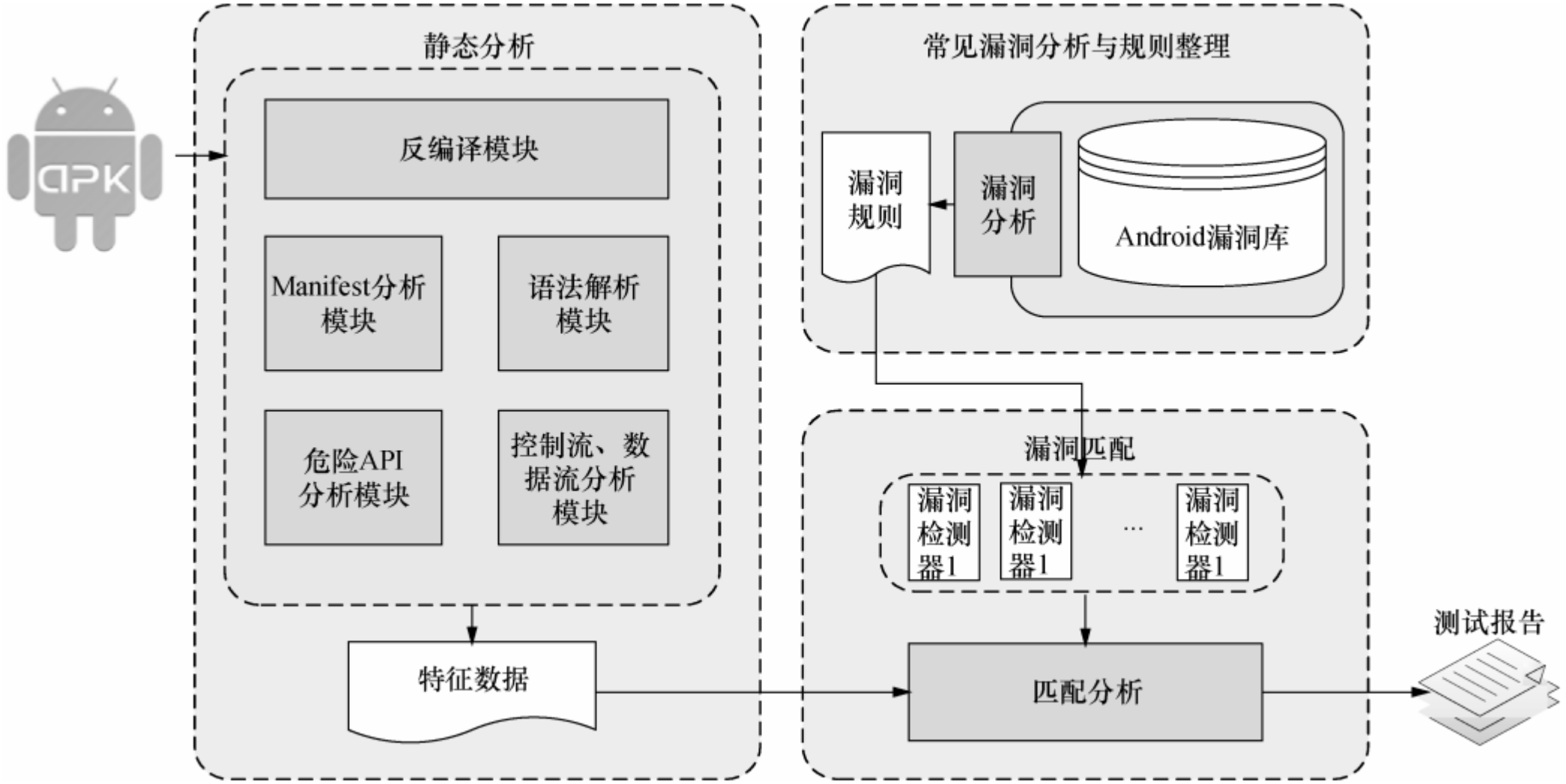
缺点：

- 仿真环境下工作，效率较低
- 自动化UI遍历效率低
- 覆盖漏洞种类少

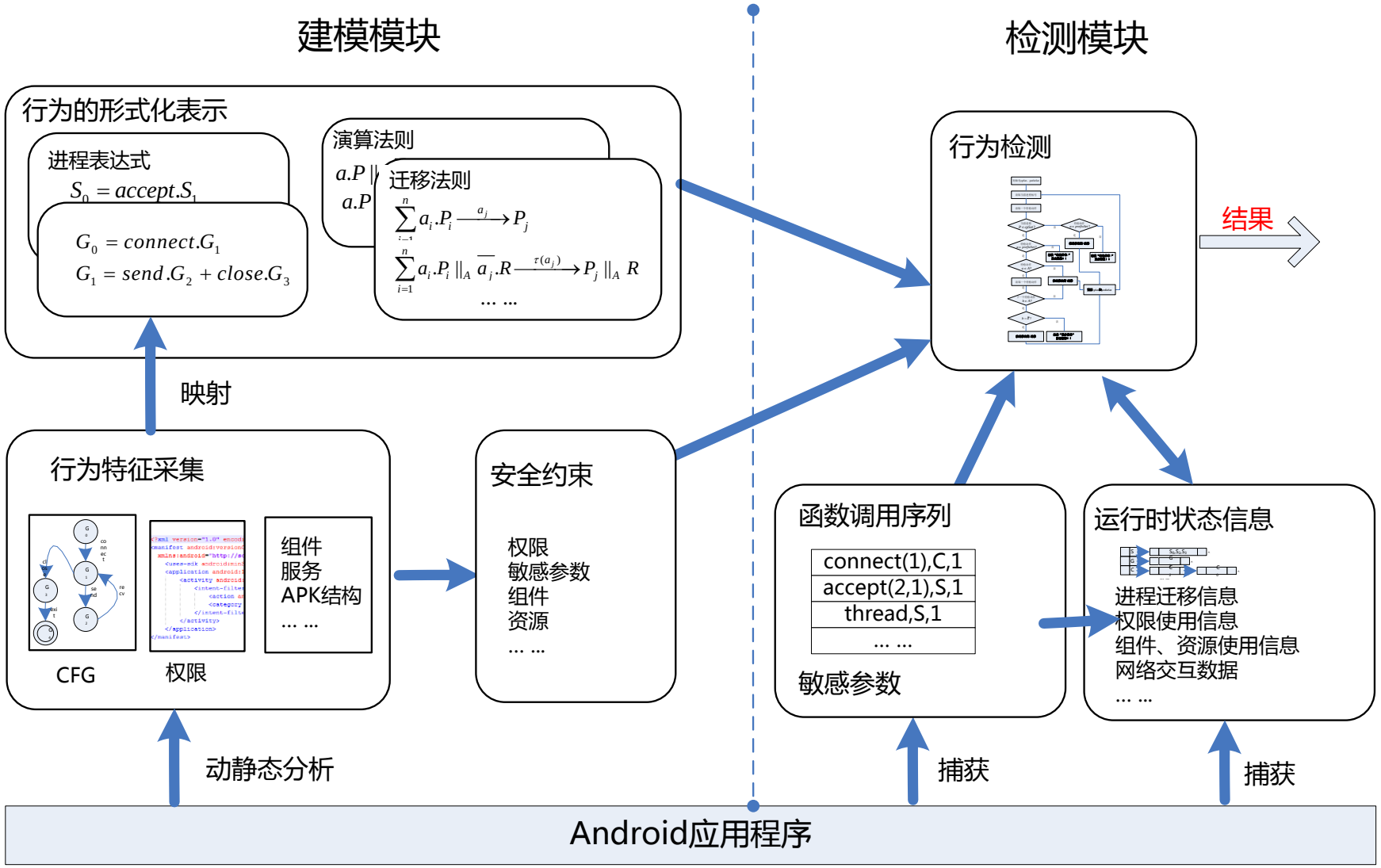
开源分析框架

工具	MobSF	AndroBugs	QARK
简介	开源移动安全测试框架	Android脆弱性分析框架	快速的Android漏洞审计工具
特点	提供可视化WEB界面 静态和动态分析	命令行操作 静态分析	可交互式命令行操作 静态分析
优点	功能完善，扩展性强，便于二次开发	检测效率高，问题定位准确	使用多种反编译并合并分析结果包含多种常见安全漏洞检测
缺点	检测时间较长	检测漏洞较少	检测漏洞较少

自动化漏洞测试架构设计



自动化漏洞测试架构设计



某金融客户 - APP应用安全检测

安全性诉求

- 敏感信息安全
- 账户安全
- 传输安全
- 应用自身安全

检测发现的痛点漏洞：

1.APK包可以进行反编译，在代码中泄露了大量的敏感URL信息和支付密钥等信息。

2.APP可以进行重打包，存在恶意代码注入风险。

3.APP对用户登录的策略没有多重验证，存在口令暴力破解风险。

4.使用了HTTP协议进行数据明文传输，存在信息泄露风险。

5.通过修改一个UserId就能获取到其它用户数据，存在越权访问风险。

解决方案：

1.对APP进行加密和加固。

2.对APP的完整性进行校验。

3.修改口令强度，并采用多重验证策略（如使用短信或图形验证码限制）。

5.对用户请求的数据包4.使用HTTPS协议进行数据传输加密。

做完整性校验，并进行多因子认证。

某银行机构 - Web端安全检测

安全性诉求

- 交易安全
- 账户安全
- 环境安全
- 传输安全
- 政策要求、行业规定

检测发现的痛点漏洞：

1.能够绕过验证码进行密码爆破。

2.任意登录用户可垂直越权获取系统所有用户信息及密码。

3.系统的密码是未经过加密后传输，可以查看到传输的账号密码等信息。

解决方案：

1.图形验证码应该从服务器端生成，并且在服务器端对验证码进行验证。

2.使用多因子验证策略对用户权限进行严格控制。

3.使用HTTPS协议进行数据传输加密。

ctrip tech 携程技术中心

ctrip tech 携程技术中心 | IT大咖说 知识分享平台

THANK YOU!

Q&A