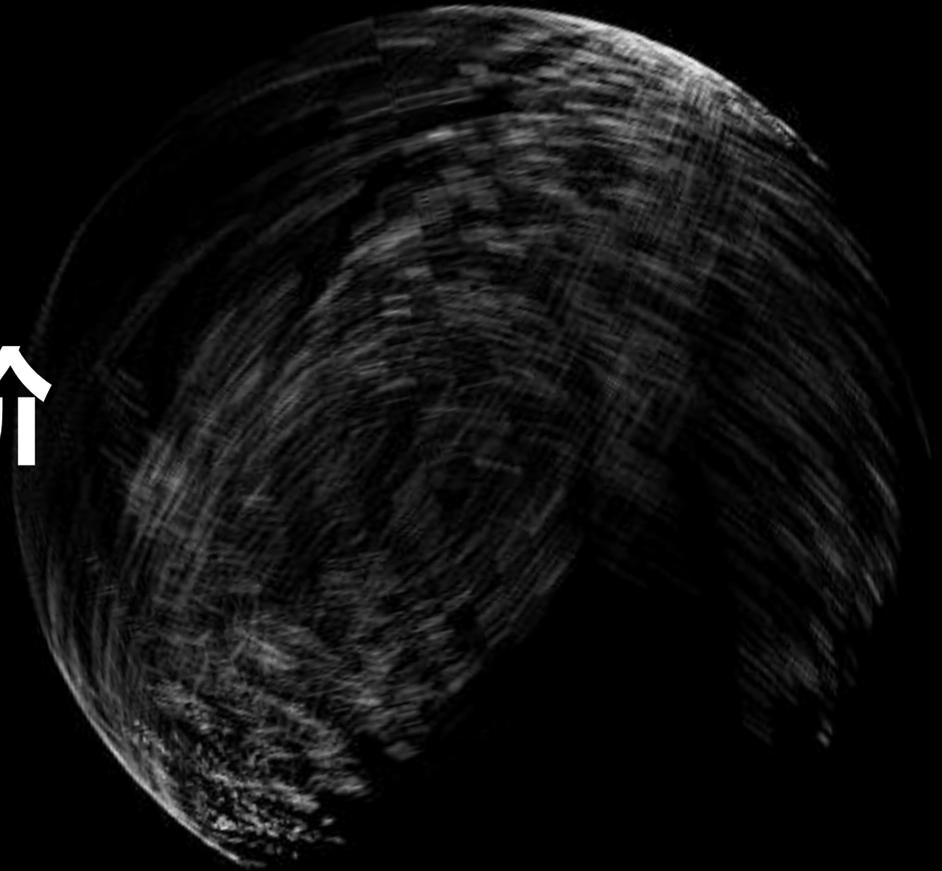


互联网+时代

Android应用安全进阶

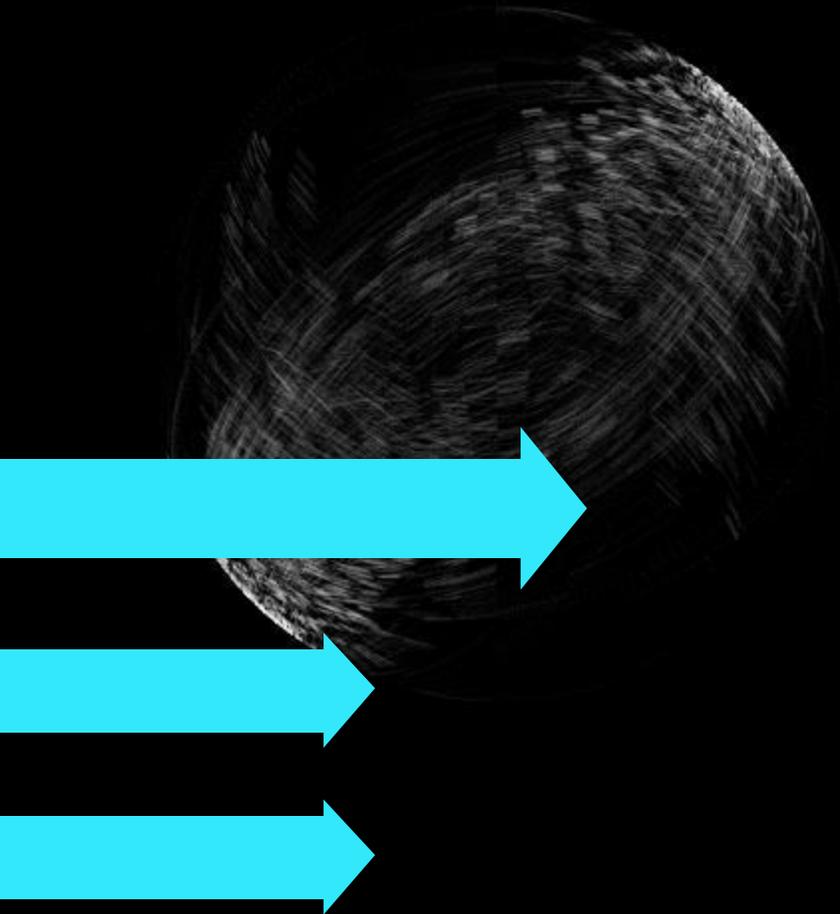
网易 卓辉





目录 Contents

01. 移动APP的安全风险
02. 移动安全进阶
03. 未知的安全风险



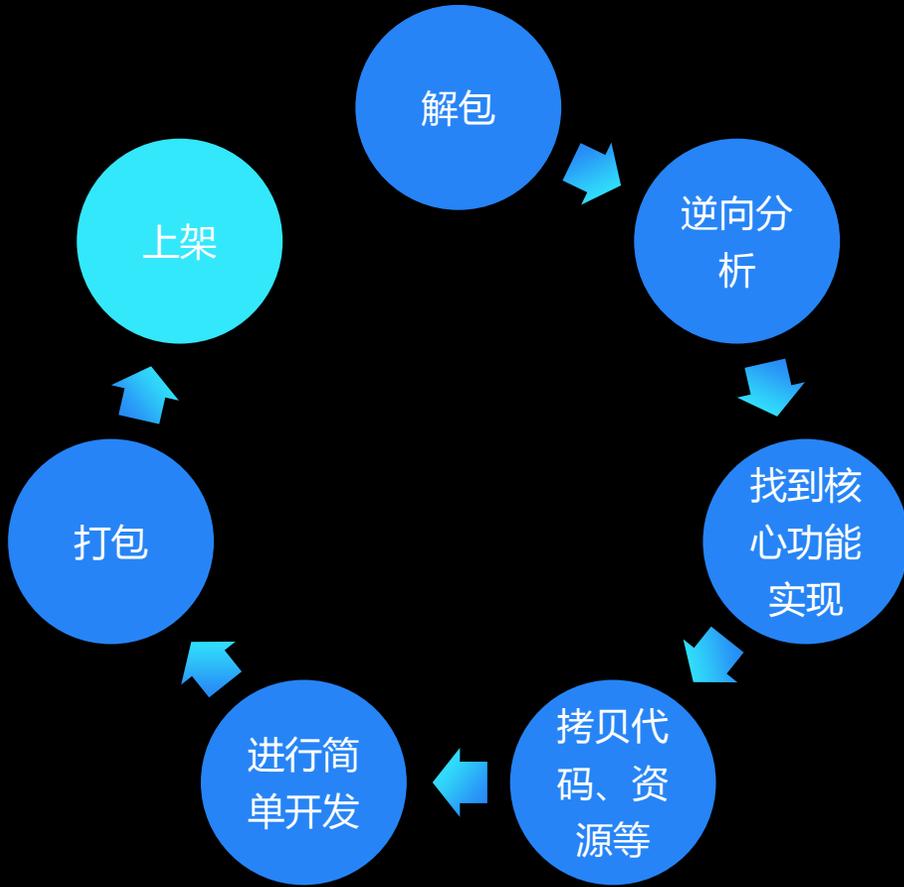
01. 移动APP的安全风险

02. 移动安全进阶

03. 未知的安全风险

山寨危险

热门应用平均有27个山寨APP，山寨应用严重危害正版应用

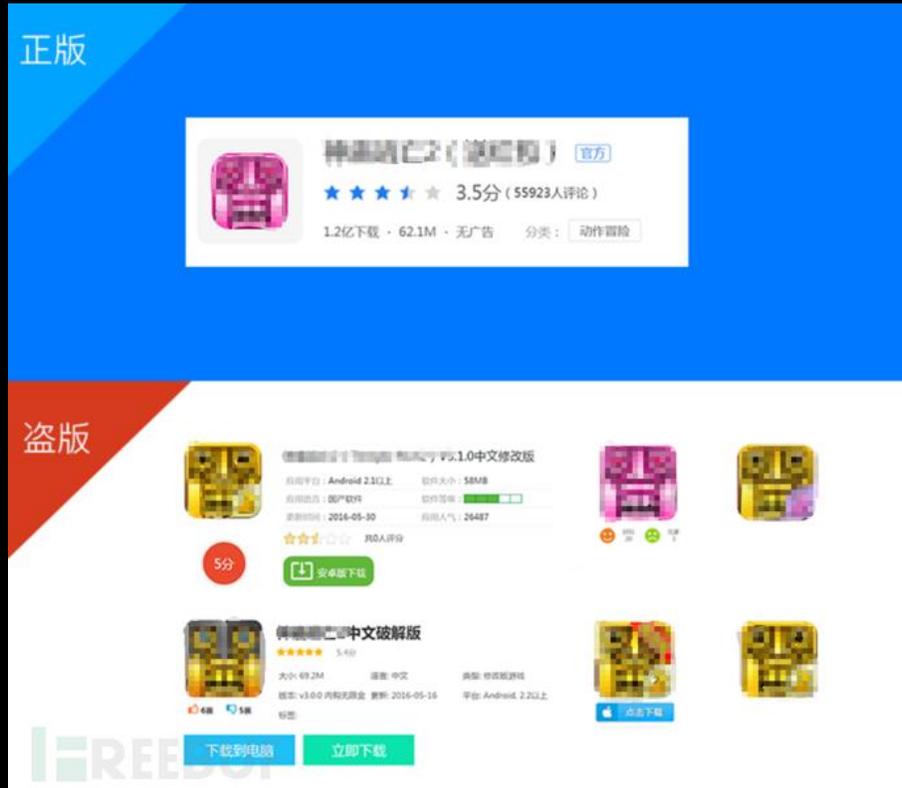


一共为您找到 146 款 "抢红包" 相关应用

	瓦力抢红包 最专业的抢红包神器！	★★★★★ 9.1分 514万次下载	安装
	极速红包王 自动抢红包神器	★★★★★ 9.9分 385万次下载	安装
	万能抢红包 2017超火微信抢红包神器	★★★★★ 9.8分 385万次下载	安装
	全自动抢红包神器 ※埋雷※盗雷※尾数	★★★★★ 9.1分 56万次下载	安装
	红包快手 红包神器自动抢红包红包外挂	★★★★★ 9.2分 478万次下载	安装
	微信伴侣-自动抢红包 抢红包斗图清垃圾用微信伴侣	★★★★★ 5.6分 276万次下载	安装
	抢红包神器 日常生活赚取零花钱必备神器	★★★★★ 8.2分 18万次下载	安装
	财神抢红包 急速秒抢QQ、微信红包神器	★★★★★ 9.9分 395万次下载	安装

146个抢红包APK

神庙逃亡被打包党二次打包



二次打包

“打包党”们通过反编译工具向应用中插入广告代码与相关配置，再在第三方应用市场、论坛发布。

- 插入自己广告或者删除原来广告
- 恶意代码, 恶意扣费、木马等
- 修改原来支付逻辑

严重危害产品和用户利益，影响公司口碑



金融、支付类本地存储数据泄露

RecNo	_id	urlid	name	value
1	1	1	actPwd	136666
2	2	2	mobile	136666
3	3	1	actPwd	498230
4	4	1	identityNo	3[redacted]02251 身份证号码
5	5	1	realName	姓名
6	6	2	authCode	104198
7	7	3	quickPayVo.cardNo	62260957107 卡号
8	8	3	quickPayVo.cardAccountName	姓名 快速支付
9	9	3	quickPayVo.mobilePhone	136666
10	10	3	quickPayVo.certificateNo	34128[redacted] 身份证号码



数据抓包，泄露用户名和密码

Web Sessions

Host	URL
appapi[redacted].com.cn	/api/Account/PostLogin
appapi[redacted].com.cn	/api/Account/PostLogin
CONNECT	tracking.miui.com:443
CONNECT	sdkconfig.ad.xiaomi.com:443
CONNECT	android.clients.google.com:443
CONNECT	www.googleapis.com:443
CONNECT	android.clients.google.com:443
CONNECT	android.clients.google.com:443
CONNECT	sp0.baidu.com:443
CONNECT	sp2.baidu.com:443
sp0.baidu.com	/5bU_dTmfKqQFm2e88IuM_a/w.gif?q=VMP
CONNECT	www.baidu.com:443
sp2.baidu.com	/8LUYsjW91Qh3otqbppnN2Djv/link?url=_C

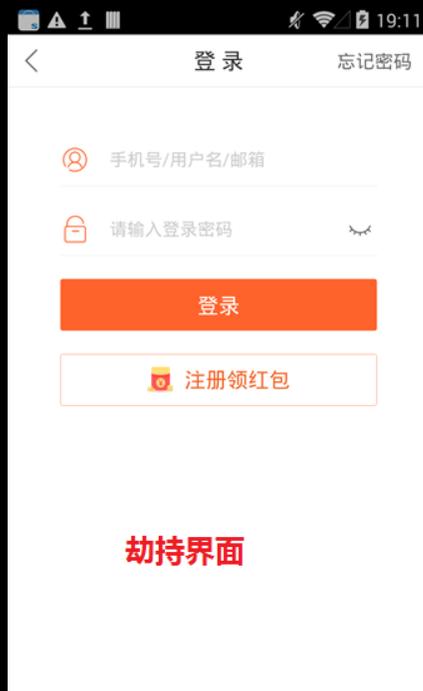
```
["d":{"E":{"868715024788592"},"P":{"123456789"},"U":{"user2017"},"FX":{"0.0"},"PY":{"0.0}}]
```



界面劫持风险



键盘记录风险

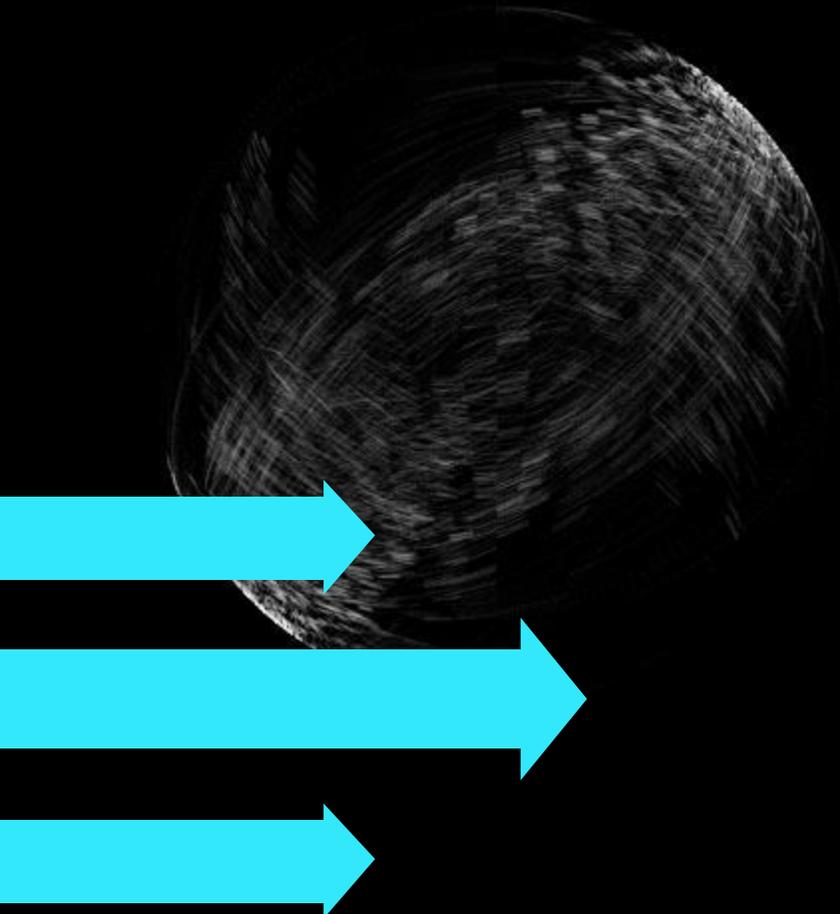


帐号、密码发送到破解者手里



```
!'}  
action=ACTION_DOWN, keyCode=KEYCODE_W, scanCode=0, metaState=0, fla  
deviceId=-1, source=0x0 }'}  
!'}  
action=ACTION_DOWN, keyCode=KEYCODE_UNKNOWN, scanCode=0, metaState=  
me=0, deviceId=-1, source=0x0 }'}  
!'}  
action=ACTION_DOWN, keyCode=KEYCODE_E, scanCode=0, metaState=0, fla  
deviceId=-1, source=0x0 }'}  
!'}  
action=ACTION_DOWN, keyCode=KEYCODE_UNKNOWN, scanCode=0, metaState=  
me=0, deviceId=-1, source=0x0 }'}  
!'}  
action=ACTION_DOWN, keyCode=KEYCODE_H, scanCode=0, metaState=0, fla  
deviceId=-1, source=0x0 }'}
```



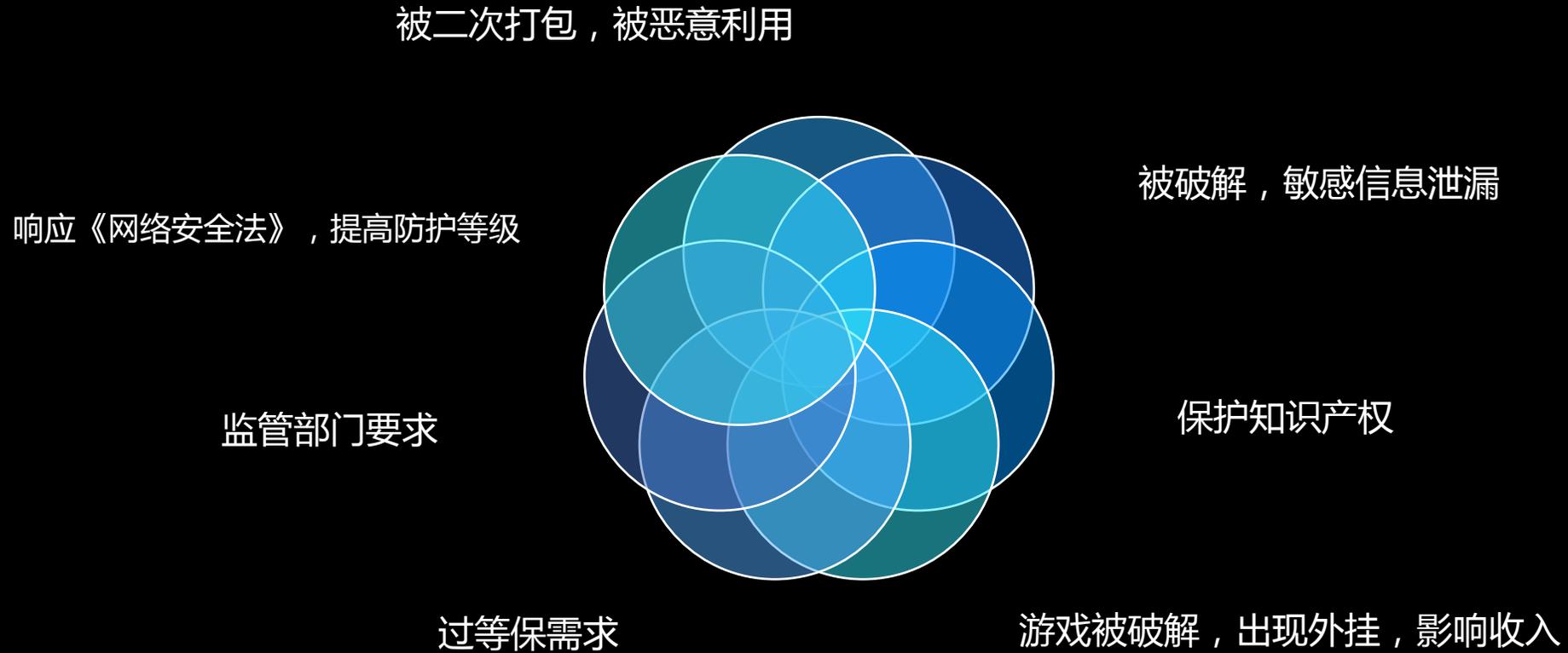


01. 移动APP的安全风险

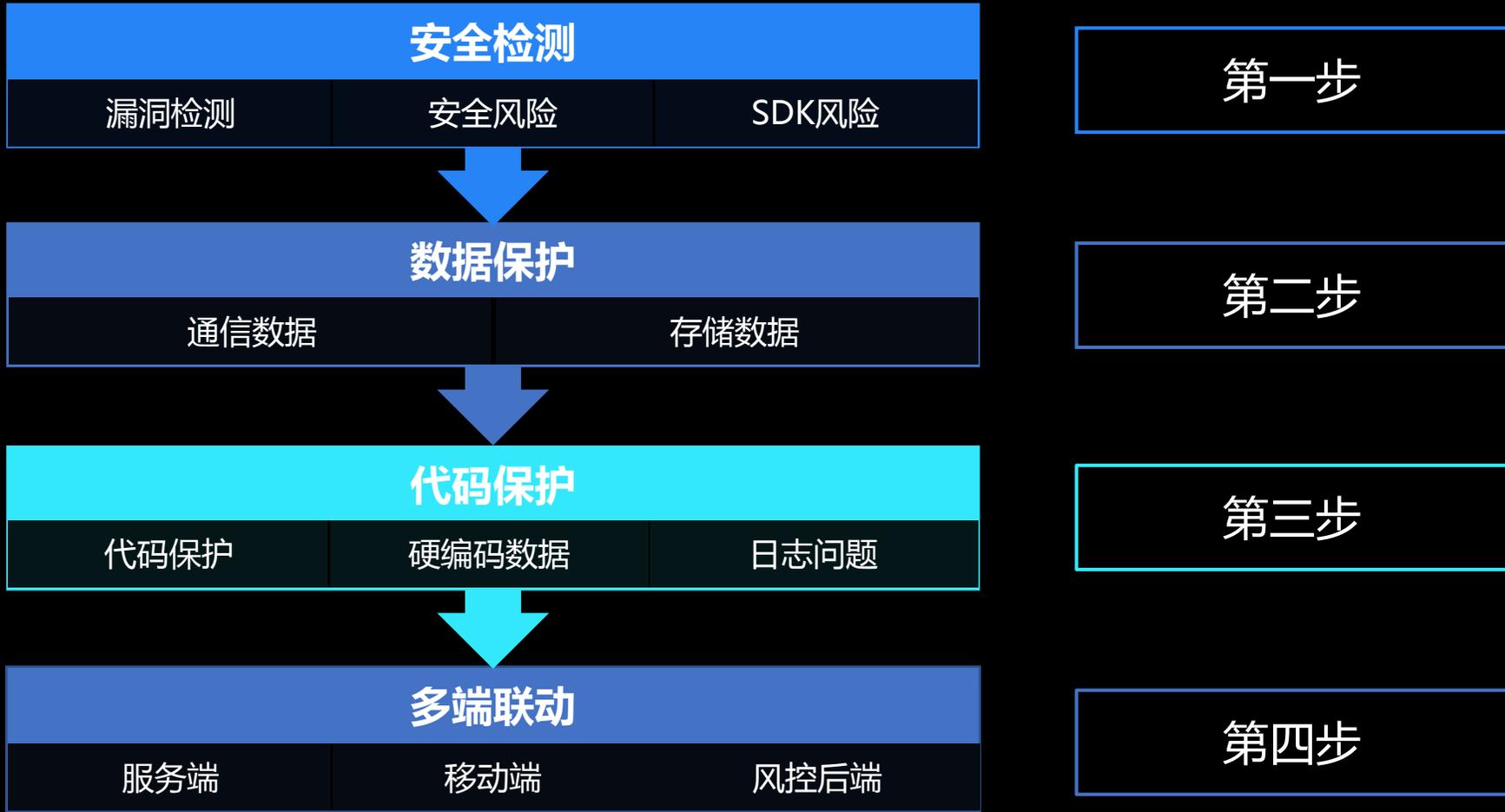
02. 移动安全进阶

03. 未知的安全风险

安全需求来源



移动安全进阶步骤



威胁类型	子类
客户端程序安全	安装包签名、客户端程序保护、应用完整性检测、组件安全、webview组件安全
敏感信息安全	数据文件、logcat日志、sqlite敏感信息明文存储、全局文件读写、敏感信息明文存储、敏感信息硬编码等。
密码软键盘安全性	键盘劫持、随机布局软键盘、屏幕录像、系统底层击键记录
安全策略设置	密码复杂度检测、账号登录限制、账户锁定策略、会话安全设置、界面切换保护、UI信息泄露、验证码安全性、安全退出、activity界面劫持
手势密码安全性	手势密码修改和取消、手势密码本地信息保存、手势密码锁定策略、手势密码抗攻击测试
通信安全	通信加密、证书有效性、关键数据加密和校验、访问控制、客户端更新安全性、短信重放攻击、没有验证SSL证书链主机名、没有验证Server证书链、忽略证书错误检测
业务功能测试	与Web测试类同
配置文件	允许调试、允许备份、Permission级别保护缺陷、activity/receiver/service公开、activity-Alias公开、provider公开、动态注册Receiver权限控制缺陷
拒绝服务	未验证Intent中数据、通用型
本地SQL注入	本地SQL注入

安全检测最主要是帮助产品规避安全风险

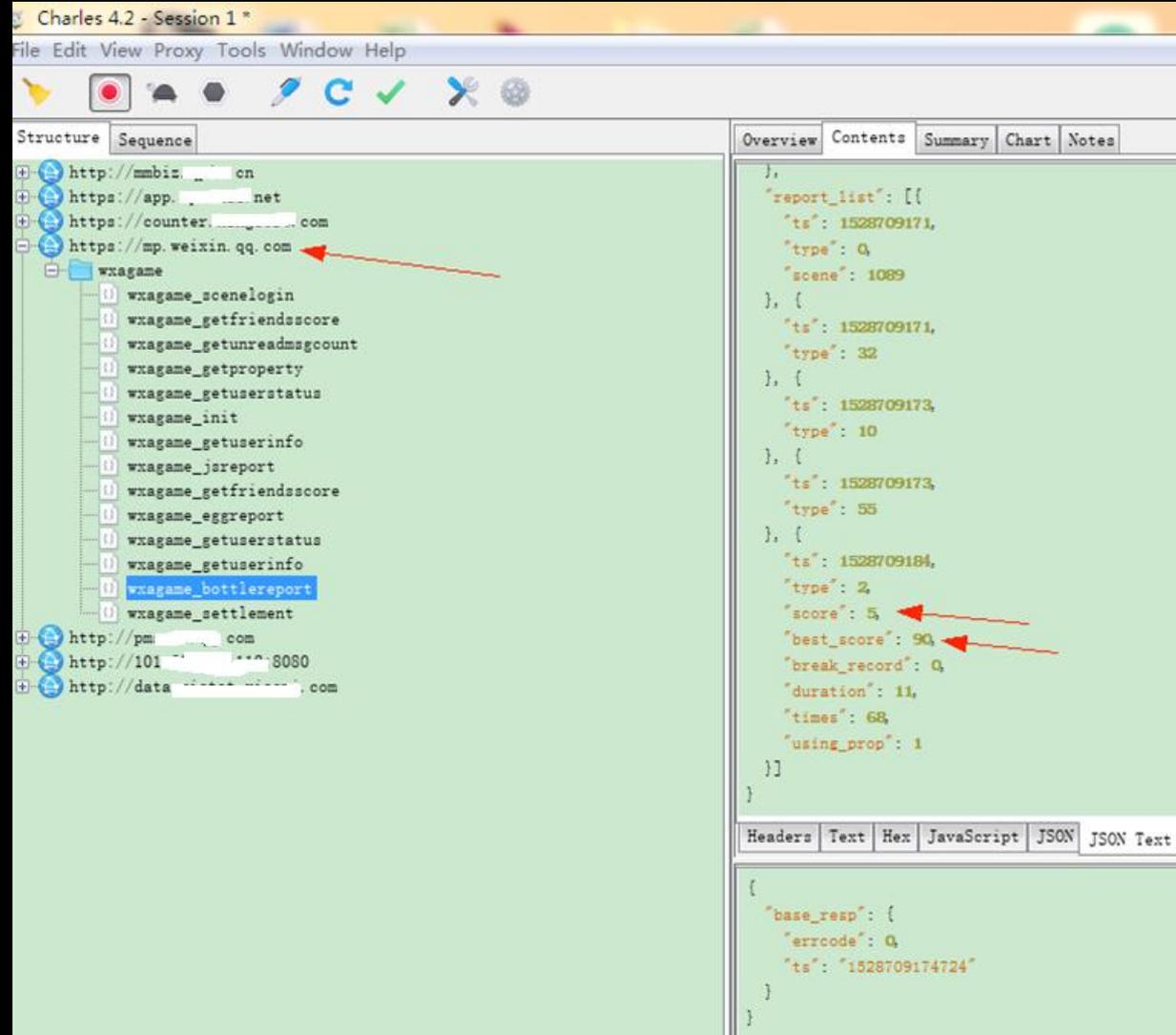
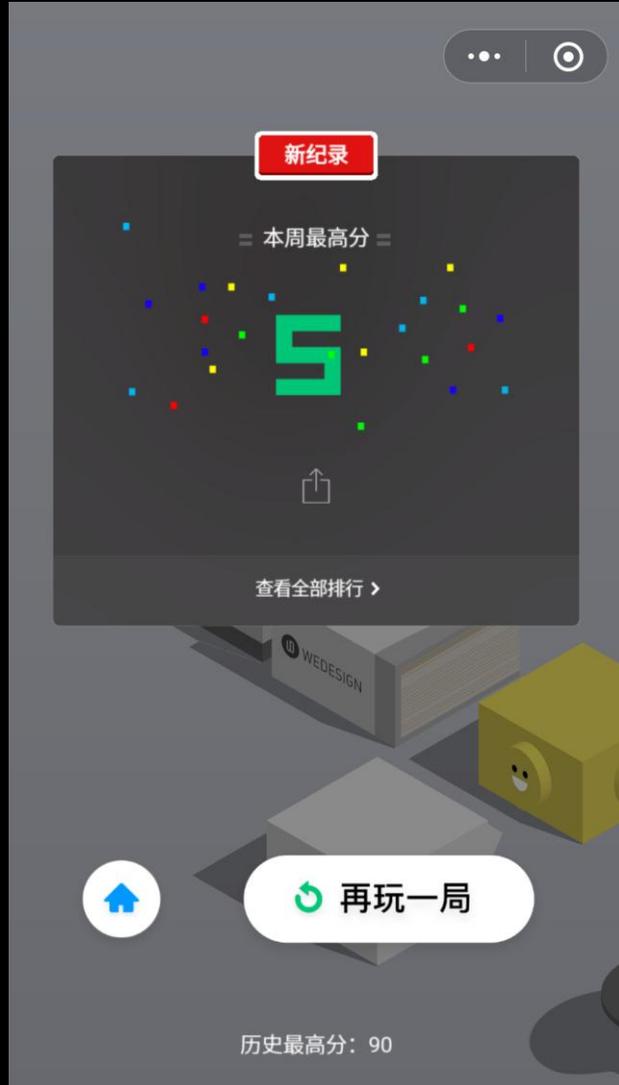
开发应该关注这些漏洞, 并想办法规避这些风险

2018年已知部分漏洞

1. ZipperDown安全漏洞
2. Janus签名漏洞
3. 应用克隆漏洞
4. RCE漏洞
5. Google Android缓冲区溢出漏洞
6. ...

最新漏洞：<http://www.cnvd.org.cn/>

数据保护-抓包



数据保护-通信风险

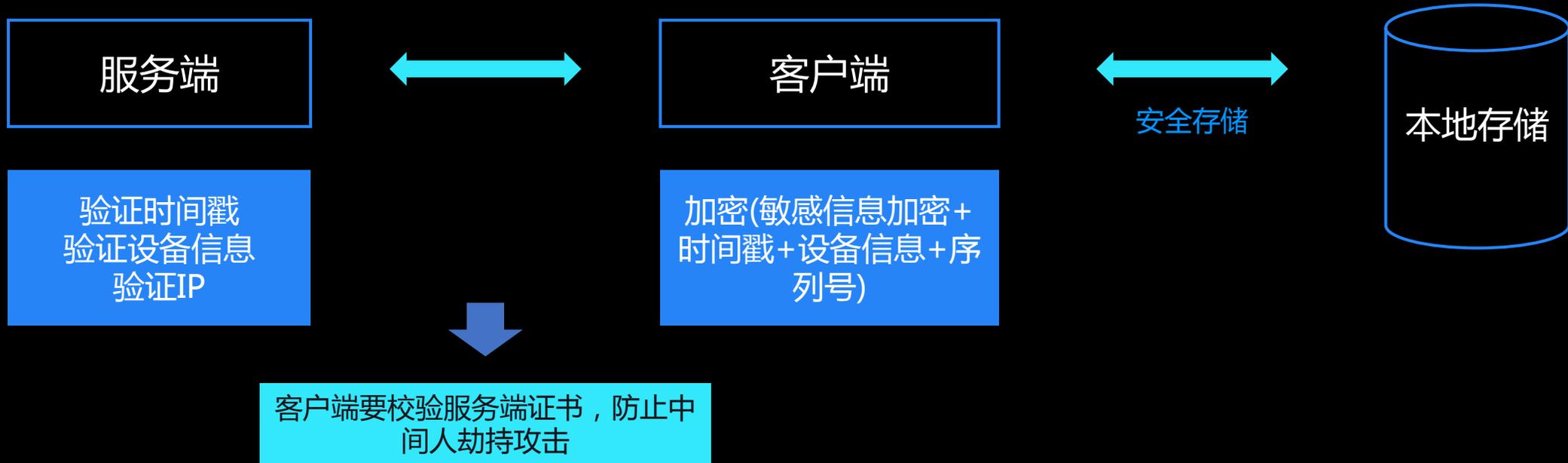
对帐号、密码做了加密处理，但这还不够



The screenshot shows the Charles 4.2 interface with a breakpoint set on a login request. The response is displayed in the JSON view, showing a successful login with fields like 'code', 'msg', 'serialSeq', 'response', 'smid', 'pushId', 'idCard', 'headImageUrl', 'mobile', 'name', 'accessToken', 'lev', and 'userId'. A blue starburst graphic with the text '登录成功' (Login Successful) is overlaid on the response.

```
{  "code": "PY0000",  "msg": "成功",  "serialSeq": "152...",  "response": {    "smid": "E1E7D9D8D9F1...ADC4188...",    "pushId": "DOA5CF6...C20E3B3",    "idCard": "3...",    "headImageUrl": "",    "mobile": "139...",    "name": "卓辉",    "accessToken": "E1E7D9D8D9F...96D7D3820B3D49",    "lev": "3",    "userId": "83...dc"  }  }
```

数据保护-怎么做？



通信数据、存储数据等重要敏感数据，要经过加密并加入校验信息

1. HTTPS没有我们想象的安全
2. 不要使用简单异或加密（不要使用自定义加密算法）
3. 本机存储数据加密并且拷贝到其它手机不能使用
4. 一机一密
5. 常用设备

开发自定义的密码输入键盘-安全键盘

不要使用手机里自带的输入法输入密码

自定义键盘布局

防截屏、录屏

防止键盘记录-记下点击坐标位置，
在底层计算出实际按键信息

代码混淆- Proguard



SDK也要混淆

代码Native化-Java转C++



Dex转到so

密钥加密: 不要简单的把密钥写在代码中



白盒加密

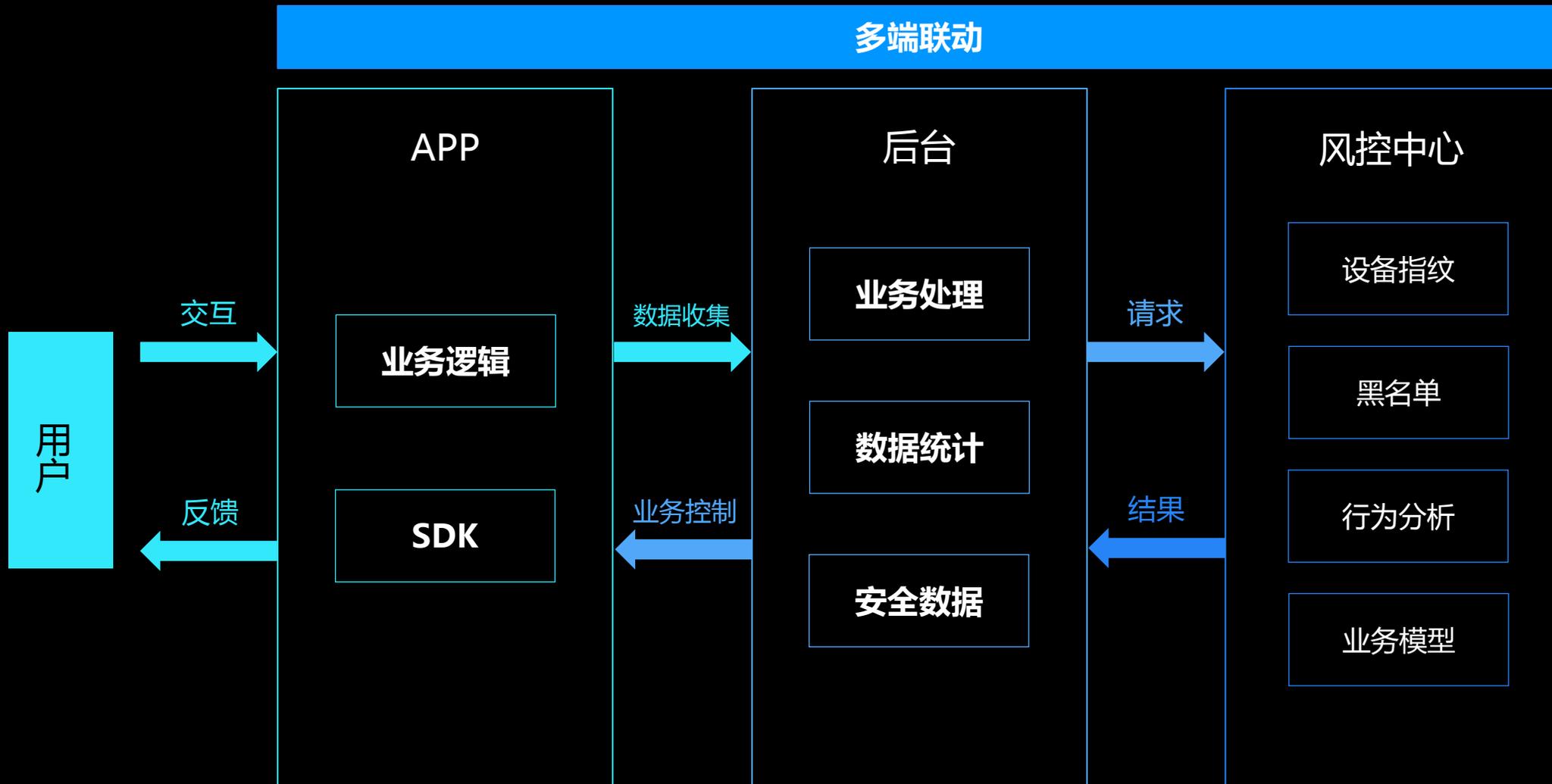
去日志化: 日志会暴露很多代码逻辑



```
-assumenosideeffects class android.util.Log{  
    public static *** v(...);  
    public static *** i(...);  
    public static *** d(...);  
    public static *** w(...);  
    public static *** e(...);  
}
```

签名校验: 防止重打包

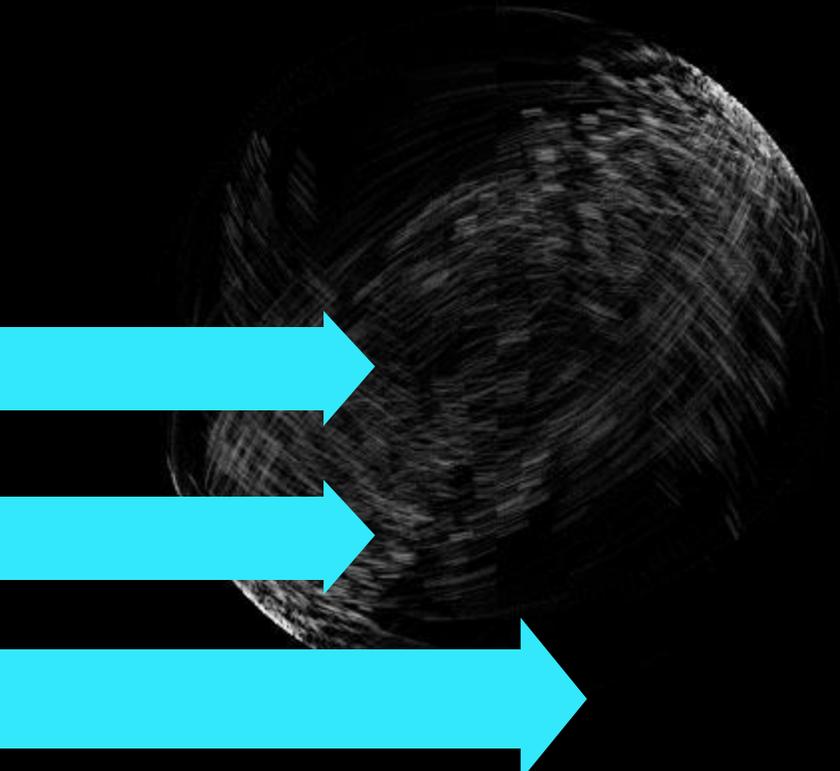




最后一步-人的安全问题

人员安全培训和安全管理，这是最容易被忽视的一块

安全中, 人是最不可控的风险因素



01. 移动APP的安全风险

02. 移动安全进阶

03. 未知的安全风险

移动安全关注的点



未来的安全风险无论对于业务，还是安全从业者，都是未知的



谢谢