数据驱动安全

2015 中国互联网安全大会
China Internet Security Conference

OAuth协议安全分析
——以Android平台为例

王 晖
@上海交通大学 LoCCS GoSSIP

关于我

- GoSSIP成员
  - 研究方向：协议分析、应用密码学、Android安全
  - 微博 @GoSSIP_SJTU
  - www.securitygossip.com

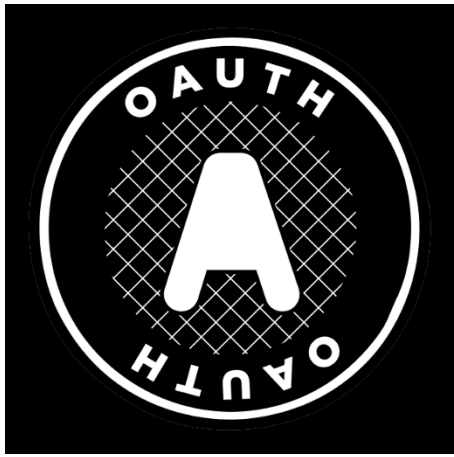
- 乌云实验室高级研究员
  - 专栏：SSL协议安全科普系列

OAUTH协议安全分析

——以Android平台为例

- 什么是OAuth协议？
- 跟我们有什么关系？

什么是OAUTH?



- 开放的授权标准（Open Authorization）
- 允许用户**授权第三方网站**访问他们存储在另外的服务提供者上的信息
- **不需要将用户名和密码**提供给第三方网站或分享他们数据的所有内容
- 2007年，OAuth 1.0
- 2012年，OAuth 2.0
- **授权** & **认证**

JD.京东 JD.COM

OAUTH跟我们有什么关系?

部分OAUTH服务提供商

服务提供商	OAuth协议版本
新浪微博	2.0
腾讯QQ	2.0
微信	2.0
支付宝	2.0
Facebook	2.0
Google	2.0
Twitter	2.0

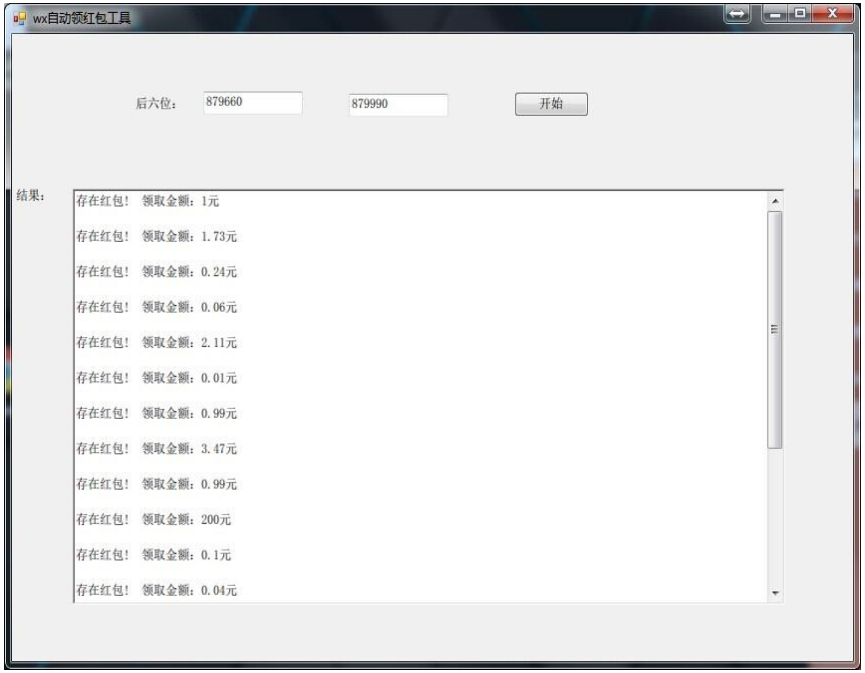
OAUTH不安全实现会怎样？——微信红包随便领

- 微信领红包URL

https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx6fa7e3bab7e15415&redirect_uri=https://wxapp.tenpay.com/v2/hybrid/www/weixin/hongbao/receive.shtml?showwxpaytitle=1&sendid=1000000000201501092047478999&channelid=1&msgtype=1&from=singlemessage&isappinstalled=0&us=**********&ver=1&sign=**********&clientversion=26000238&devicetype=android-19&pass_ticket=**********&timeguid=14207873040300.4459930493030697&response_type=code&scope=snsapi_base&state=STATE&connect_redirect=1#wechat_redirect

- **WooYun-2015-90898**

OAUTH不安全实现会怎样？——微信红包随便领



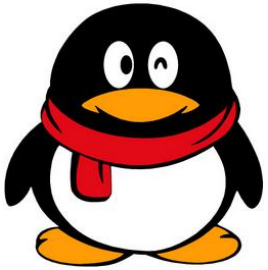
大纲

- Android平台OAuth实现的安全问题
- OAuth协议简介
- Android平台中OAuth实现的特性
- 对Android平台中OAuth实现的安全审计
- 案例分析

ANDROID平台OAUTH实现的安全问题

- 国内15家主流OAUTH服务提供商，14家存在至少一种安全问题



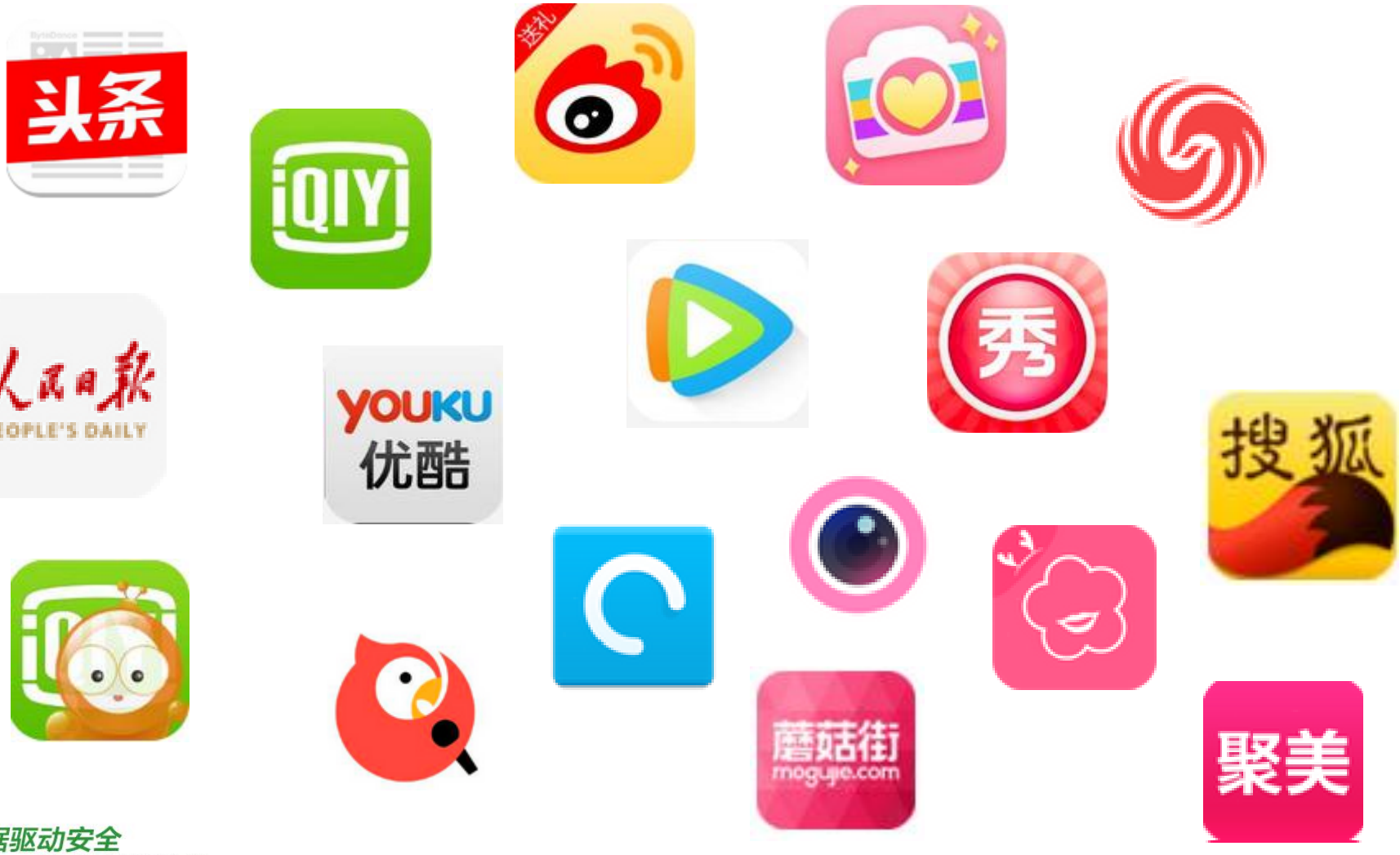




国内主流服务提供商OAUTH实现特点概况

SPs	Installs	Authorization grant	User-agent	RP app authentication	Enforced HTTPS
Sina Weibo	100-500 million	Auth code/Implicit	W/A	yes	yes
Tencent Weibo	10-50 million	Modified implicit	W/A	no	no
Qzone	100-500 million	Modified implicit	W/A/B	no	no
QQ	1-1.5 billion	Modified implicit	W/A/B	no	no
Wechat	1-1.5 billion	Auth code	A	yes	yes
Youdao Note	10-50 million	1.0a/Auth code	W/A	yes	W no/A yes
Evernote	10-50 million	1.0	W	no	yes
Yixin	10-50 million	Auth code	W/A	yes	no SSL
Douban	5-10 million	Auth code	W	no	yes
Renren	50-100 million	Auth code/Implicit	W/A	no	W no/A yes
Kaixin	10-50 million	Auth code/Implicit/Password	W	no	no SSL
Baidu	100-500 million	Auth code/Implicit	W	no	no
Taobao	0.5-1 billion	Auth code/Implicit/Password	W/A	no	no
Laiwang	10-50 million	Auth code	A	no	no
Alipay	100-500 million	Auth code/Implicit	W/A	no	no

ANDROID平台OAUTH实现的安全问题

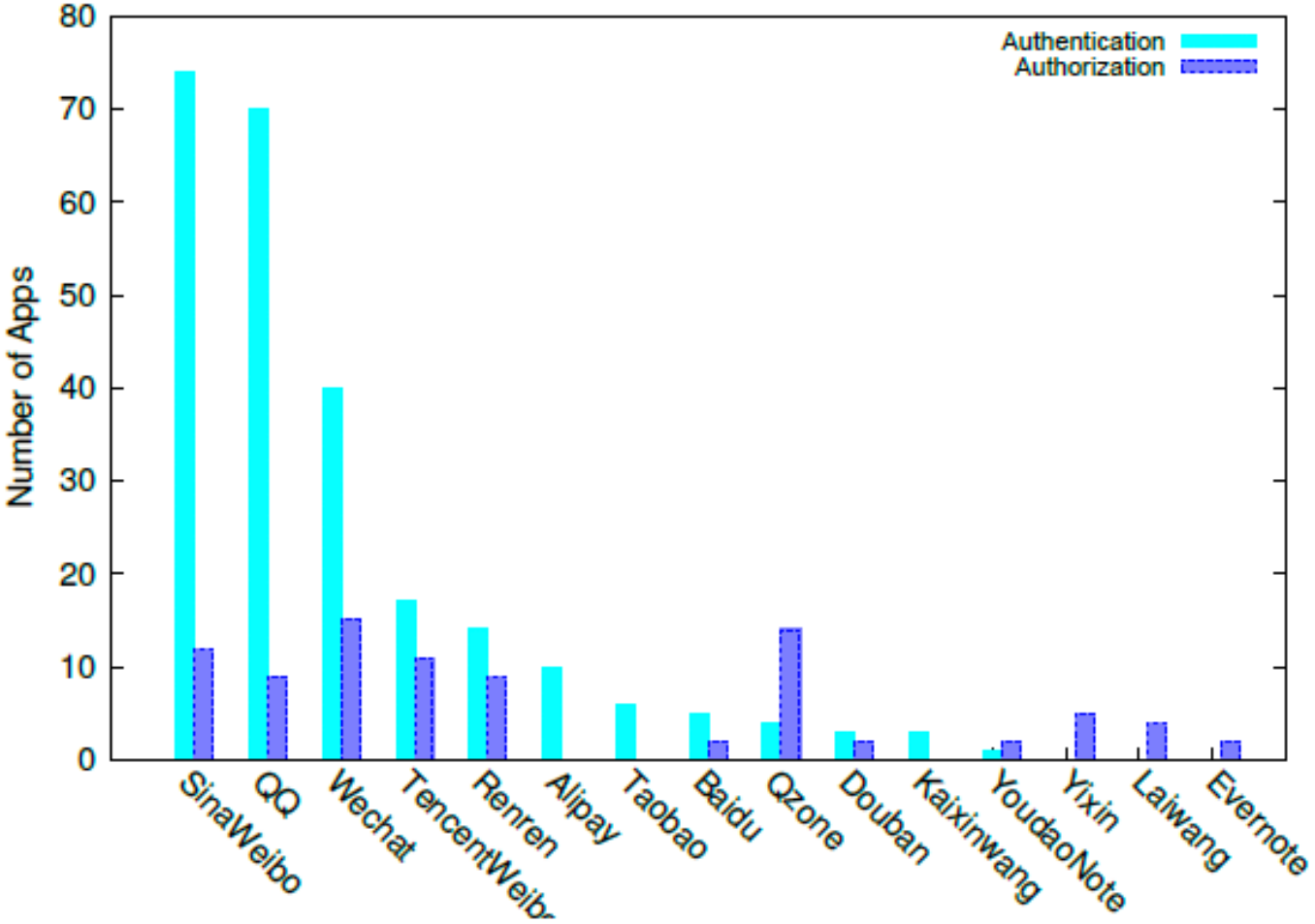
- 应用市场TOP 100的应用，84个使用OAUTH，81个存在安全问题



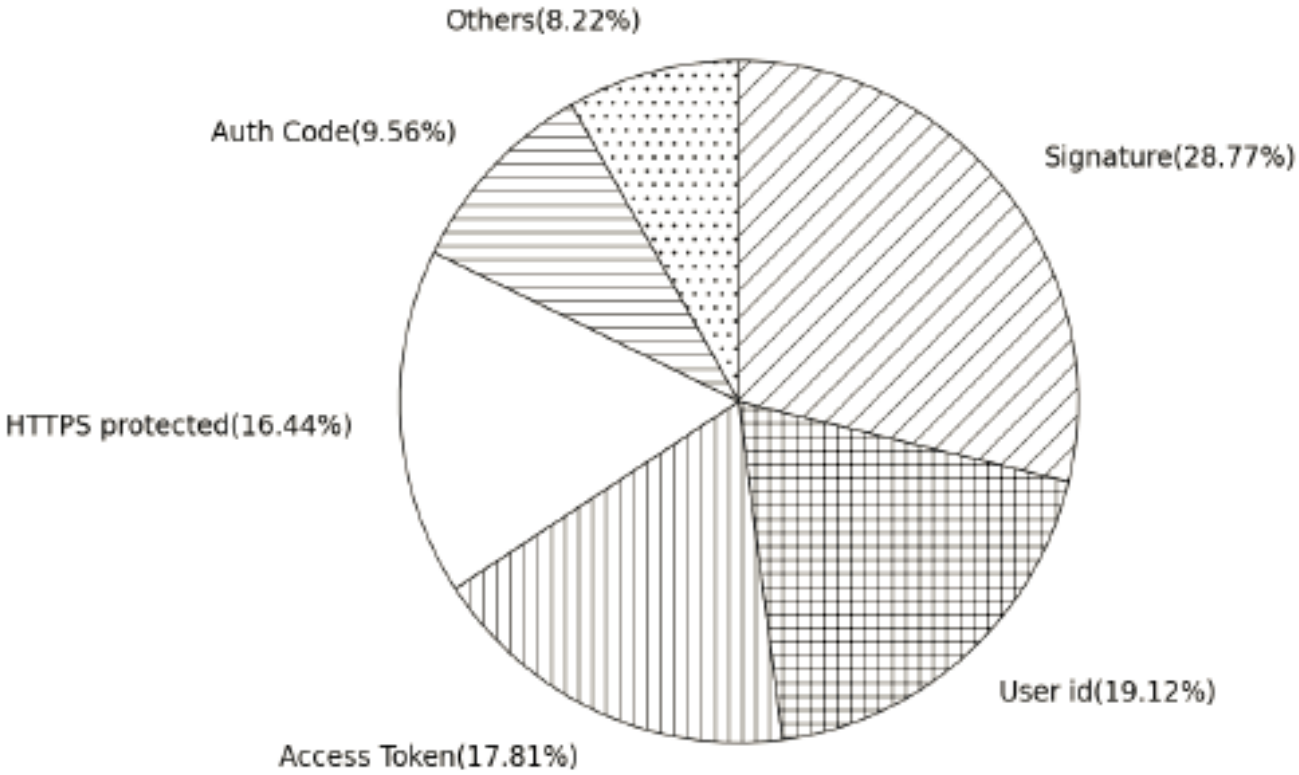
主要漏洞类型

- 不安全的用户代理（V1）
- 缺乏协议参与者身份认证（V2）
- 不安全的信息传输（V3）
- 不安全的秘密管理（V4）
- 不正确的服务器端参数校验（V5）
- 不正确的认证凭据（V6）

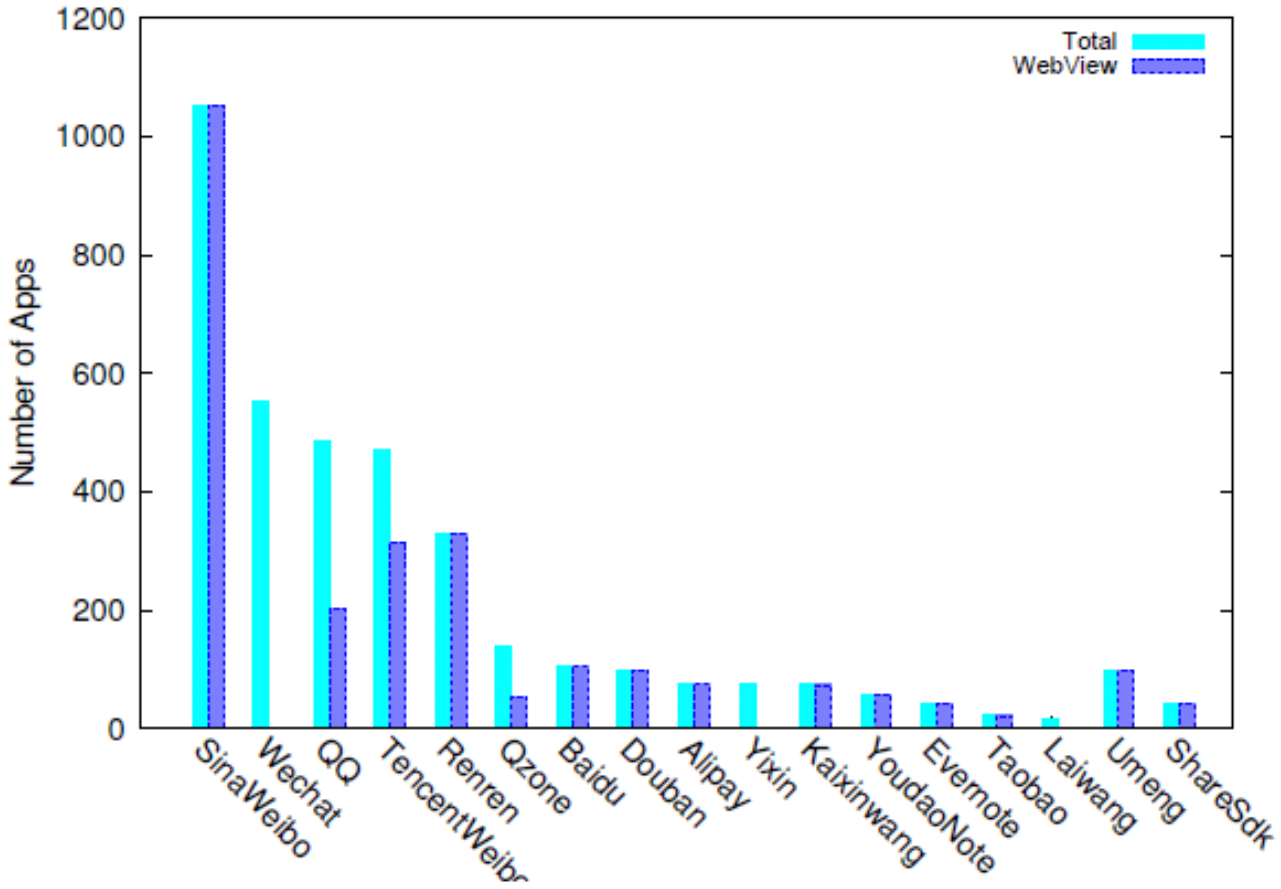
TOP 100 APP使用OAUTH服务授权和认证数量



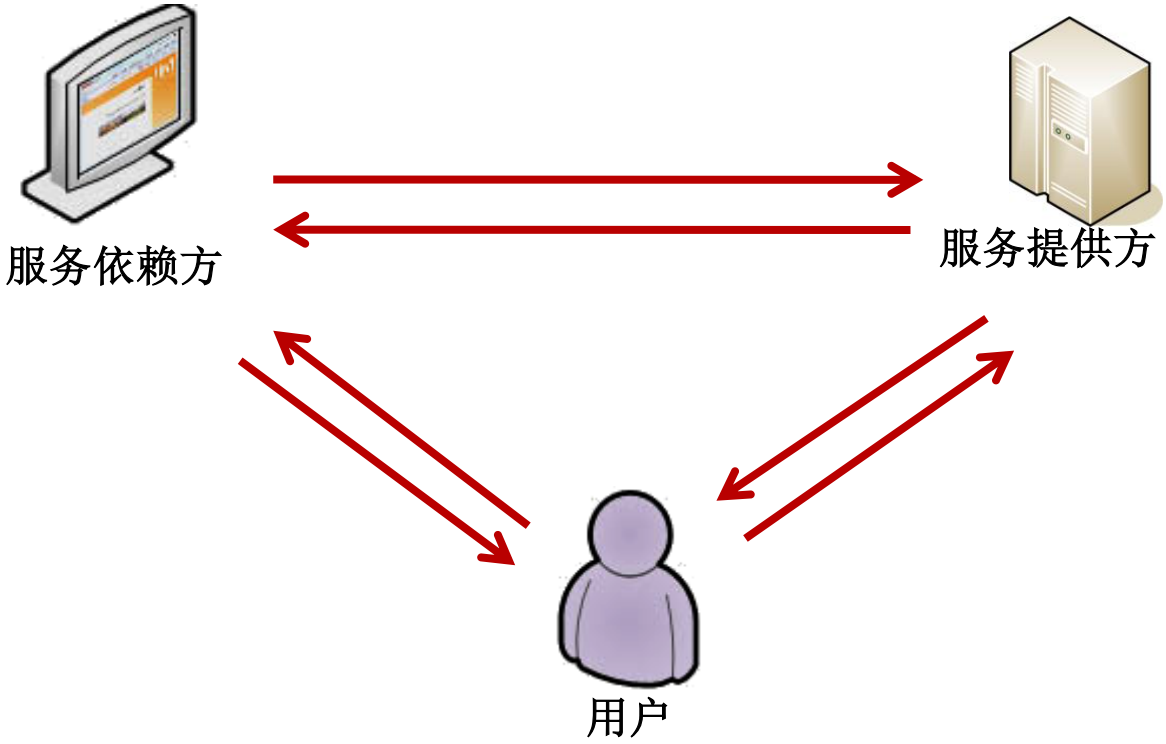
TOP 100 APP使用OAUTH认证用户方式



- 分析了4,151个应用，1,372个使用了OAUTH，86.4%存在安全问题

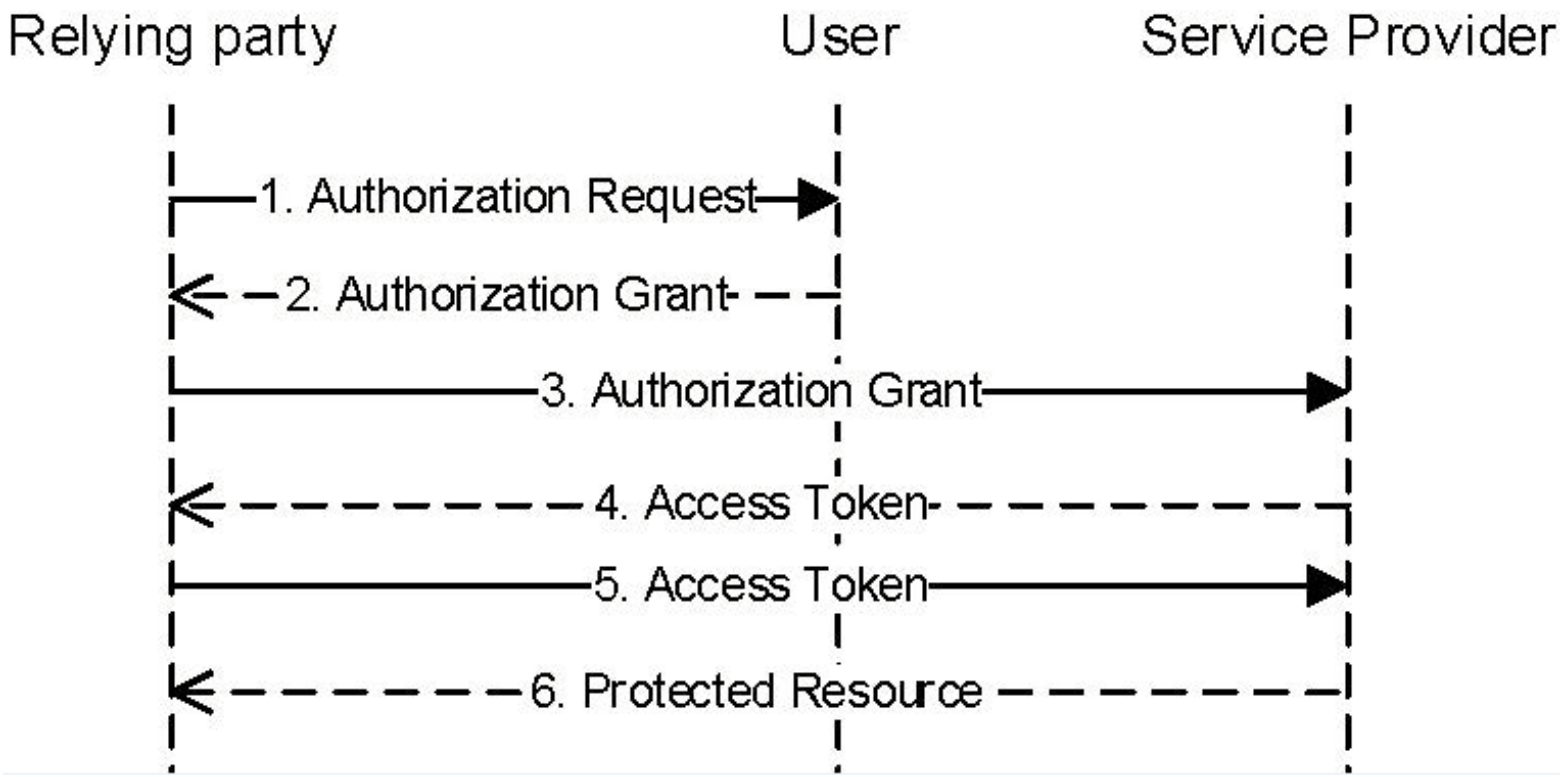


OAUTH协议简介



- 服务依赖方：Relying Party (RP) ，也称为Consumer
- 服务提供方：Service Provider (SP) ，也称为Identity Provider/ Authorization Server & Resource Server

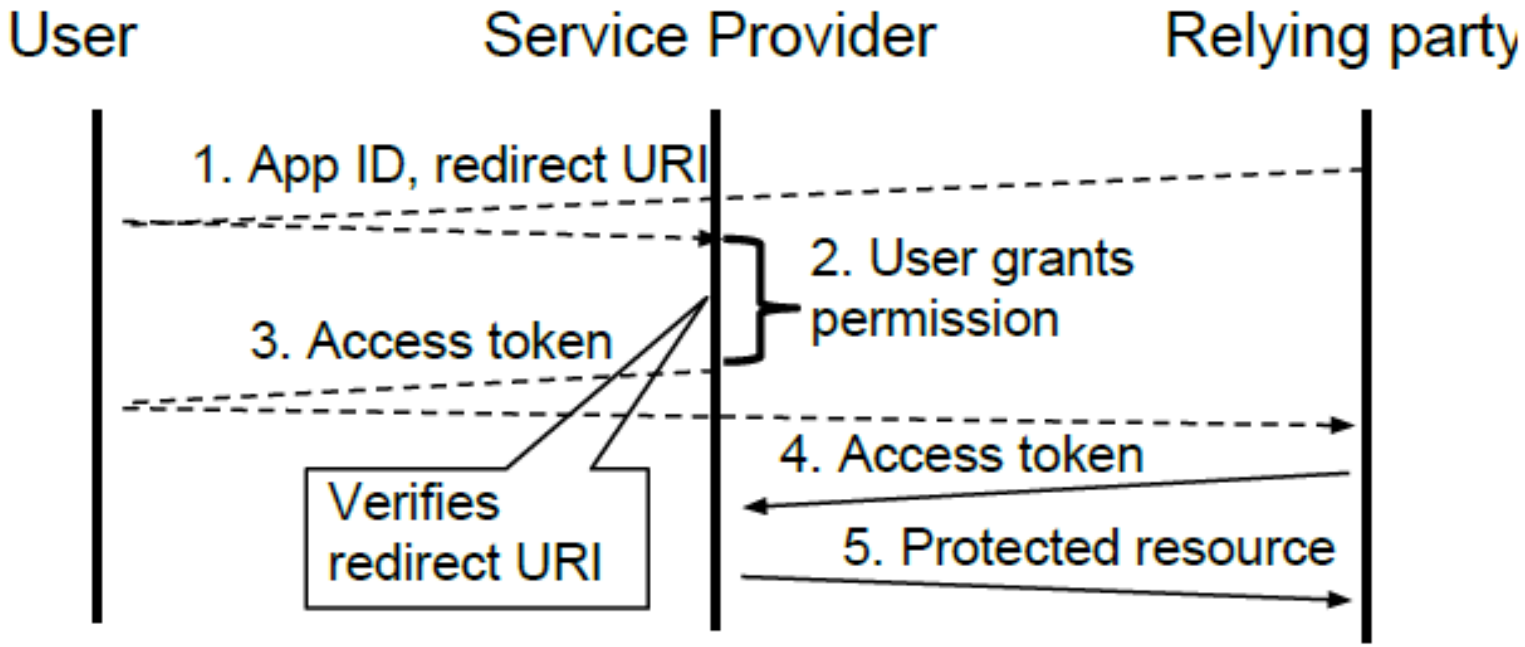
OAUTH抽象工作流


Relying party
User
Service Provider
1. Authorization Request
2. Authorization Grant
3. Authorization Grant
4. Access Token
5. Access Token
6. Protected Resource

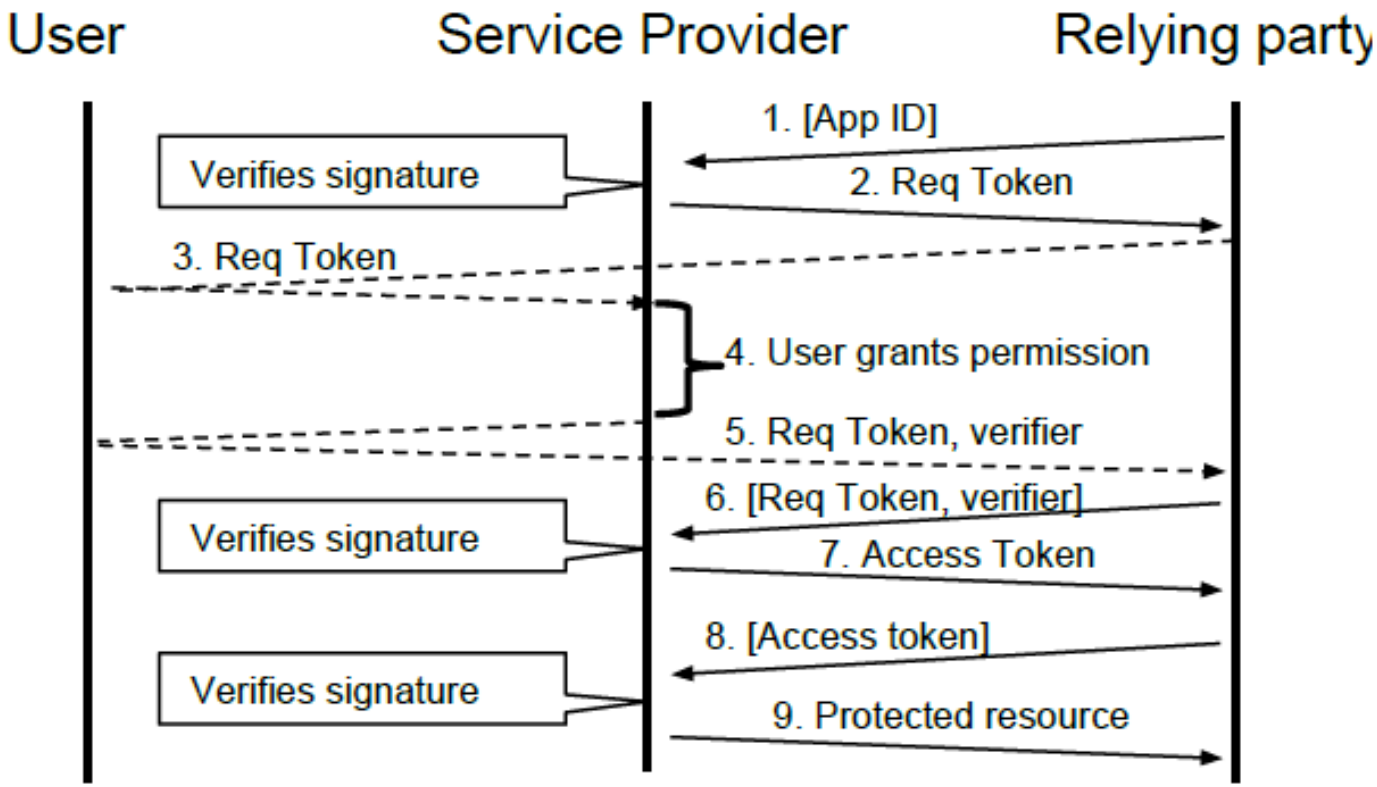
OAUTH 授权许可类型:

- OAuth 1.0/1.0a
- OAuth 2.0
  - 授权码许可 （authorization code grant）
  - 隐式授权 （implicit grant）
  - 资源拥有者密码凭据授权 （resource owner password credential grant）
  - 客户端凭据授权 （client credential grant）

OAUTH 2.0 隐式授权

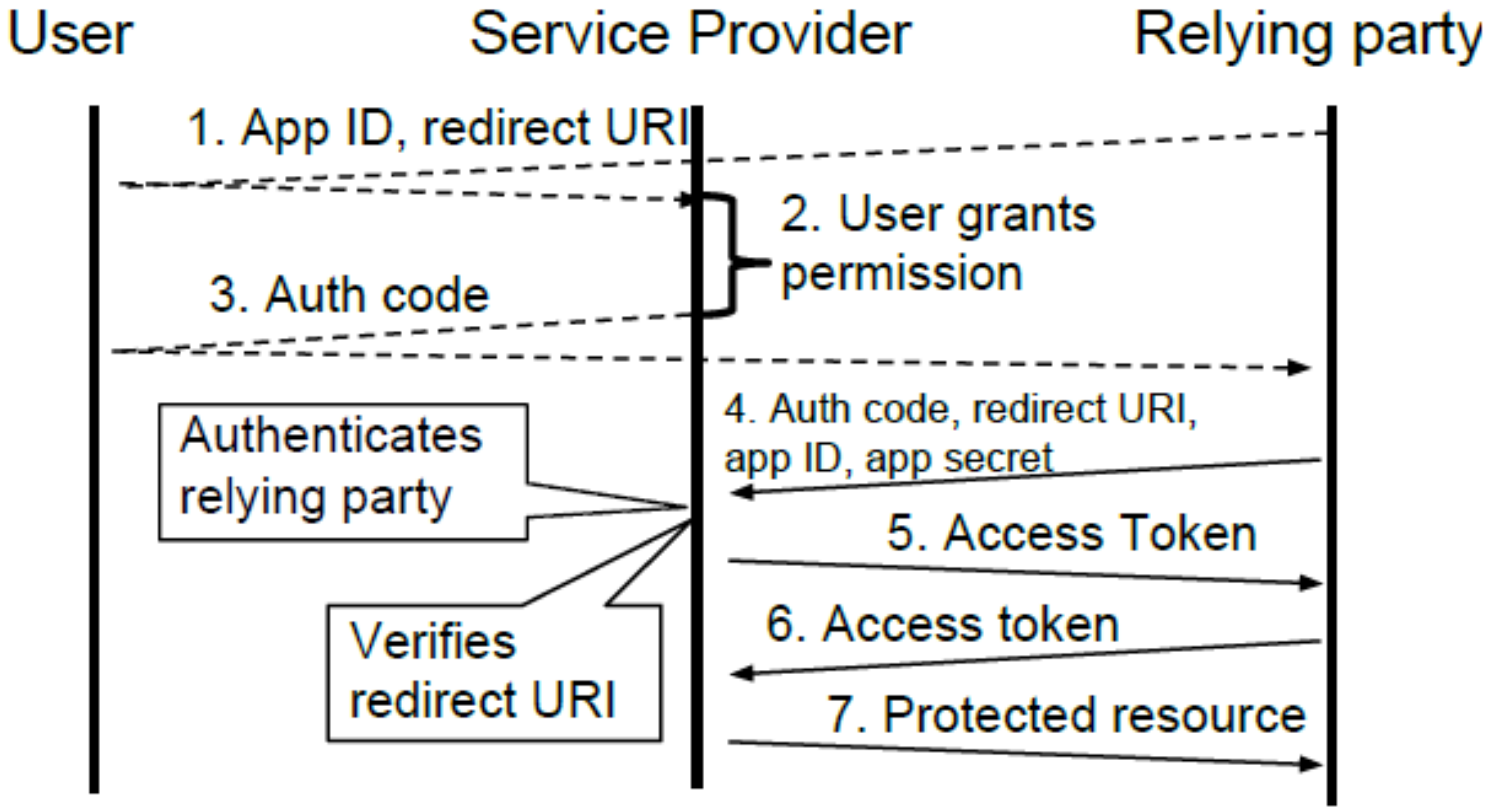


OAUTH 1.0

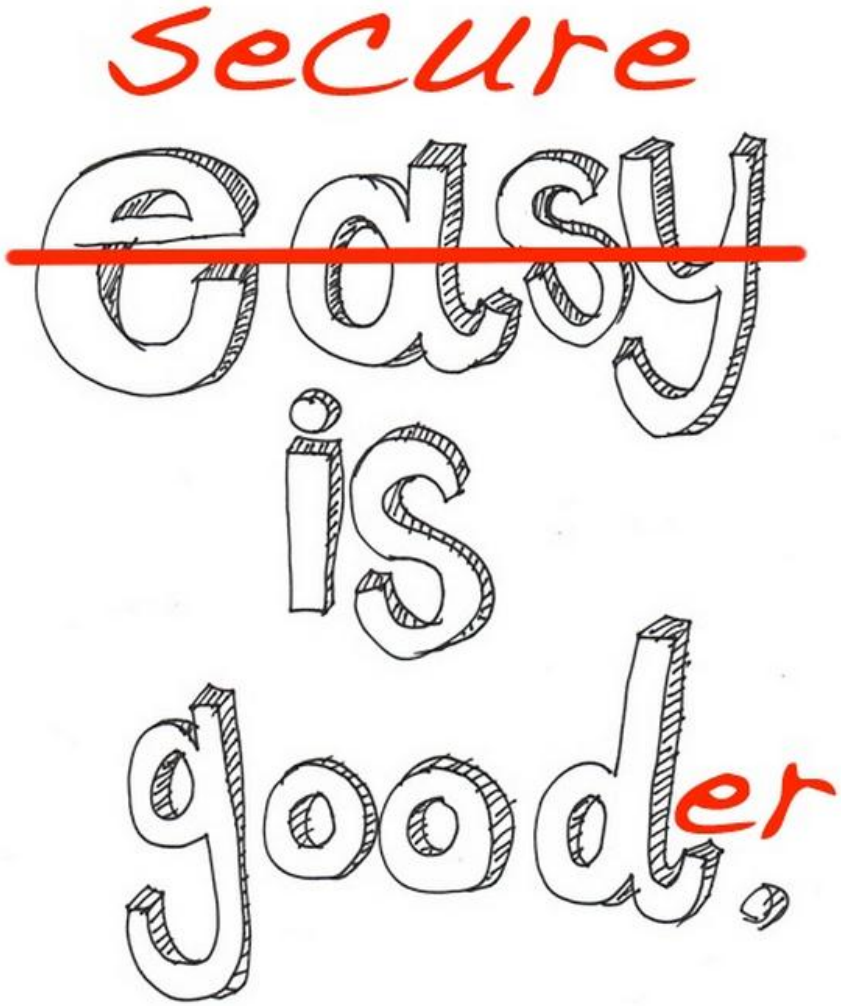

User
Service Provider
Relying party
1. [App ID]
Verifies signature
2. Req Token
3. Req Token
4. User grants permission
5. Req Token, verifier
6. [Req Token, verifier]
Verifies signature
7. Access Token
8. [Access token]
Verifies signature
9. Protected resource

OAUTH协议的工作原理

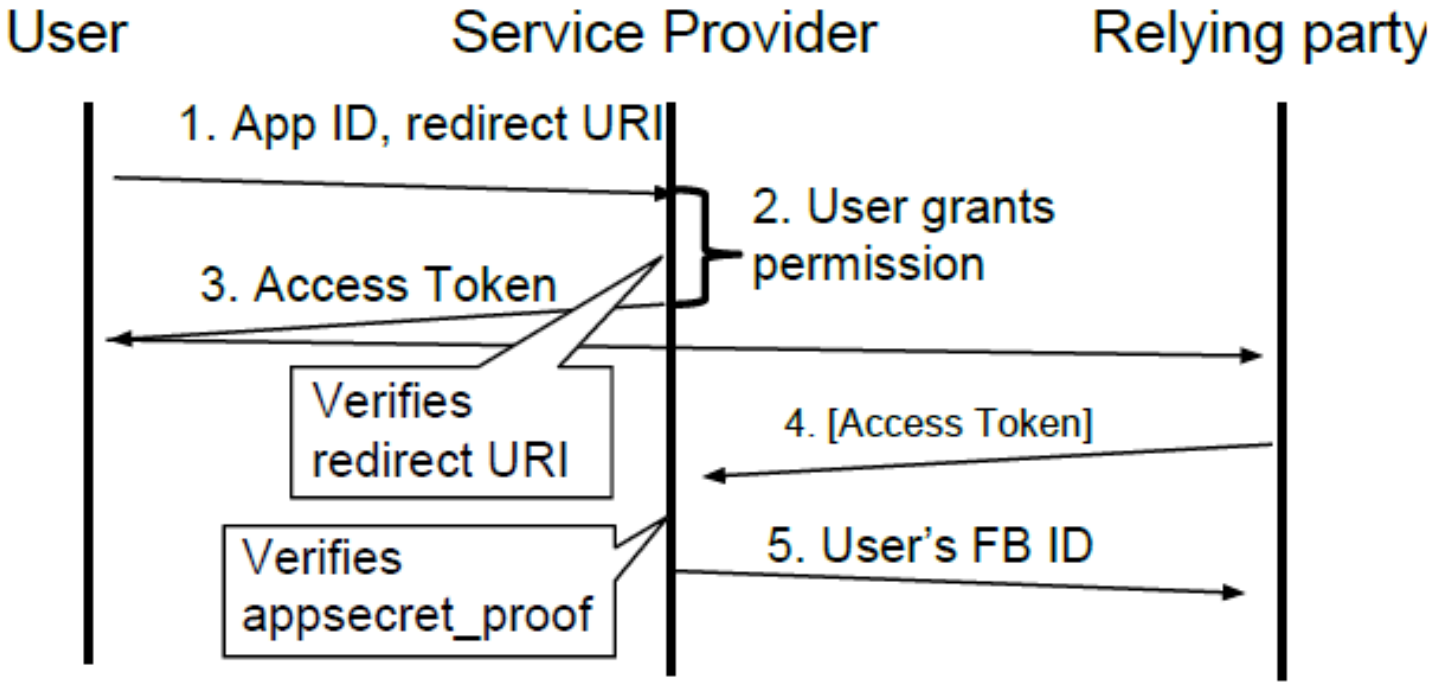
OAUTH 2.0 授权码许可



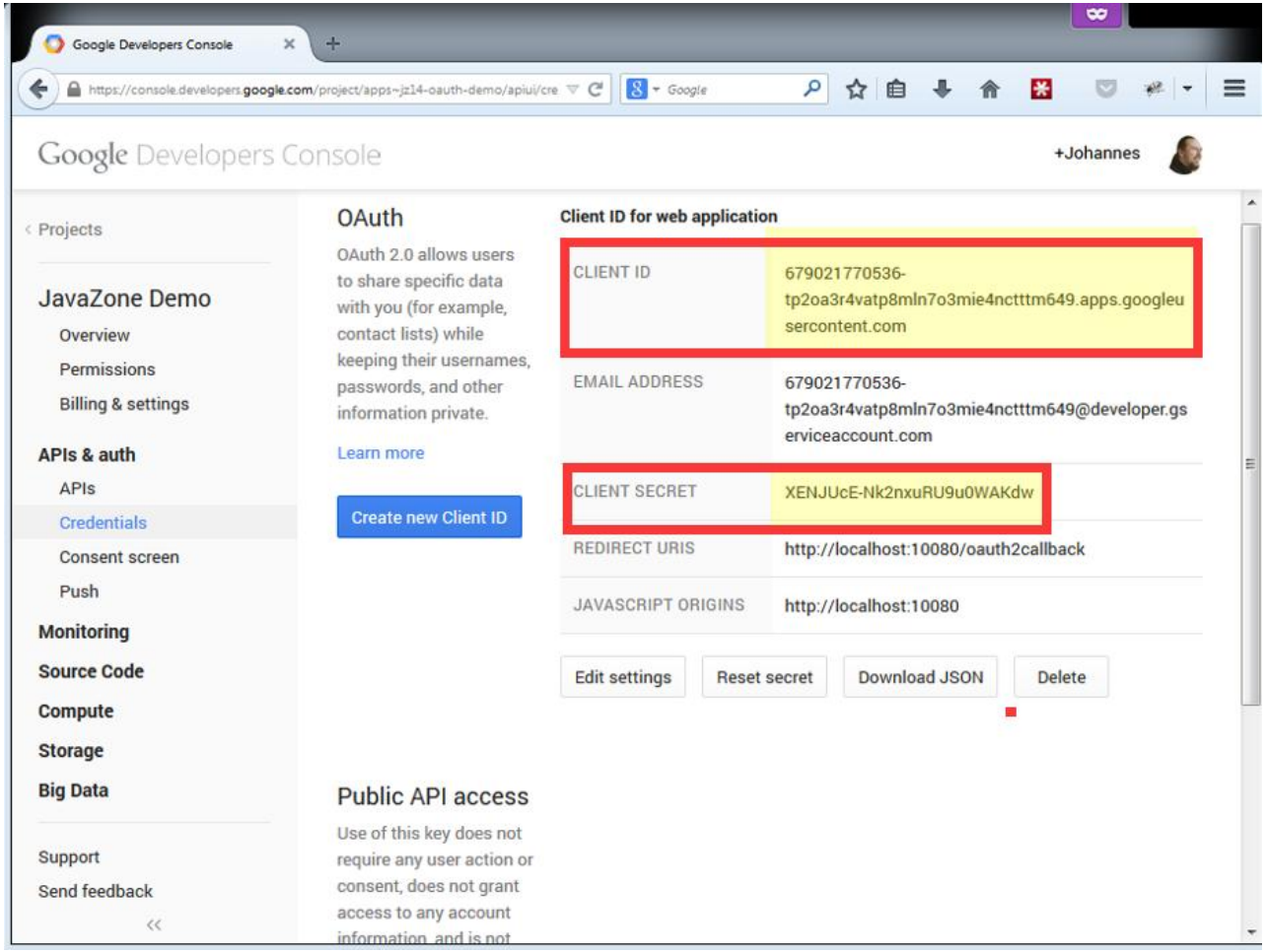
OAUTH协议的工作原理



一种安全的OAUTH工作流（FACEBOOK）



RP注册应用（WEB）


Google Developers Console
https://console.developers.google.com/project/apps~jz14-oauth-demo/apiui/cre
Google Developers Console
+Johannes
Projects
JavaZone Demo
Overview
Permissions
Billing & settings
APIs & auth
APIs
Credentials
Consent screen
Push
Monitoring
Source Code
Compute
Storage
Big Data
Support
Send feedback
OAuth
OAuth 2.0 allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private.
Learn more
Create new Client ID
Client ID for web application
CLIENT ID
679021770536-tp2oa3r4vatp8mln7o3mie4nctttm649.apps.googleusercontent.com
EMAIL ADDRESS
679021770536-tp2oa3r4vatp8mln7o3mie4nctttm649@developer.gserviceaccount.com
CLIENT SECRET
XENJUcE-Nk2nxuRU9u0WAKdw
REDIRECT URIS
http://localhost:10080/oauth2callback
JAVASCRIPT ORIGINS
http://localhost:10080
Edit settings
Reset secret
Download JSON
Delete
Public API access
Use of this key does not require any user action or consent, does not grant access to any account information, and is not

RP注册应用（MOBILE）

OAUTH 2.0 资源拥有者密码凭据授权


```
POST /oauth2/token?client_id=6704a9c60214b9196baad033c866821f&client_secret=b53fefdd7d4ee21e3903d0cb10b43f57&username=[redacted]&password=[redacted]&grant_type=password HTTP/1.1
Accept-Encoding: gzip
Connection: close
Accept-Language: zh-CN,zh;
User-Agent: xiami mobile/Android 5.1.1 build101
Host: passport.alipay.com
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
```

值得注意的安全敏感点

- 重定向URI, 访问令牌及授权码验证
- 访问令牌、授权码的生成
- 秘密管理
- 授权提示
- HTTPS保护

WEB平台和ANDROID平台OAUTH实现的差异性

- 不同的用户代理
- 不同的重定向机制
  - Web平台使用HTTP 302状态码
  - Android平台中使用Intent机制
- RP、SP及用户代理的身份验证
- 应用的客户端逻辑
- 密钥/秘密管理

攻击者模型

- 网络攻击者
- 恶意服务依赖方应用
- 重打包恶意服务提供商应用
- 重打包恶意服务依赖方应用
- Android设备中的其它恶意应用

ANDROID平台中的用户代理

- WebView
- Service Provider app
- Native system browser

Type	RP app validation	Isolation between user-agent and RP
WebView	√	√
SP app	X	X
System browser	X	√

针对WEBVIEW的攻击

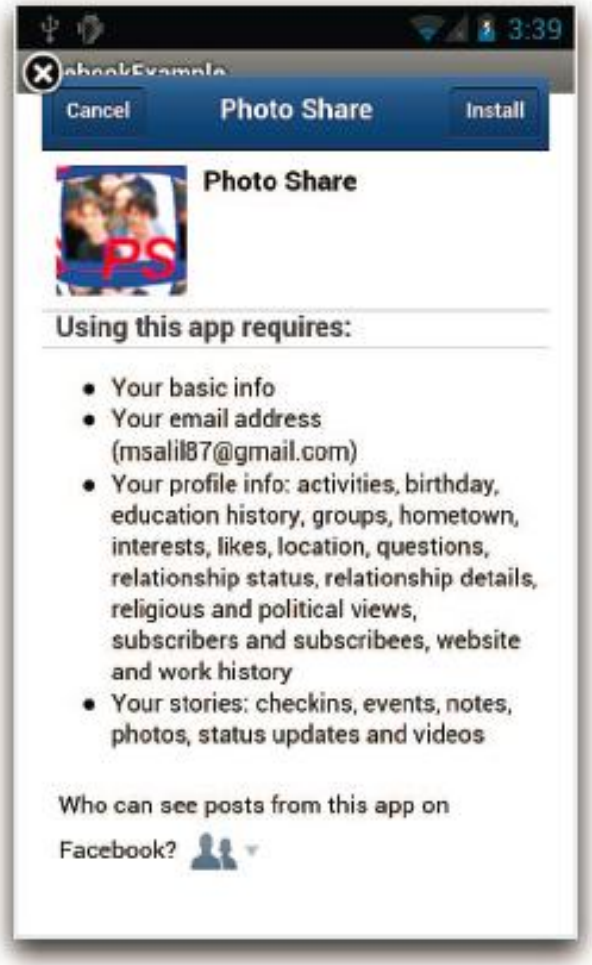
- 允许在WebView中使用JavaScript，setJavaScriptEnabled(true)
- 在WebView中注册事件句柄
  - 能够对不同事件进行响应
  - 能够检测WebView中的活动，如 “onLoadResource()” 和 “onPageFinished()”
- 在WebView中注入一个本地(Java) 对象 ，允许这个对象的方法被JS访问 “addJavascriptInterface()”

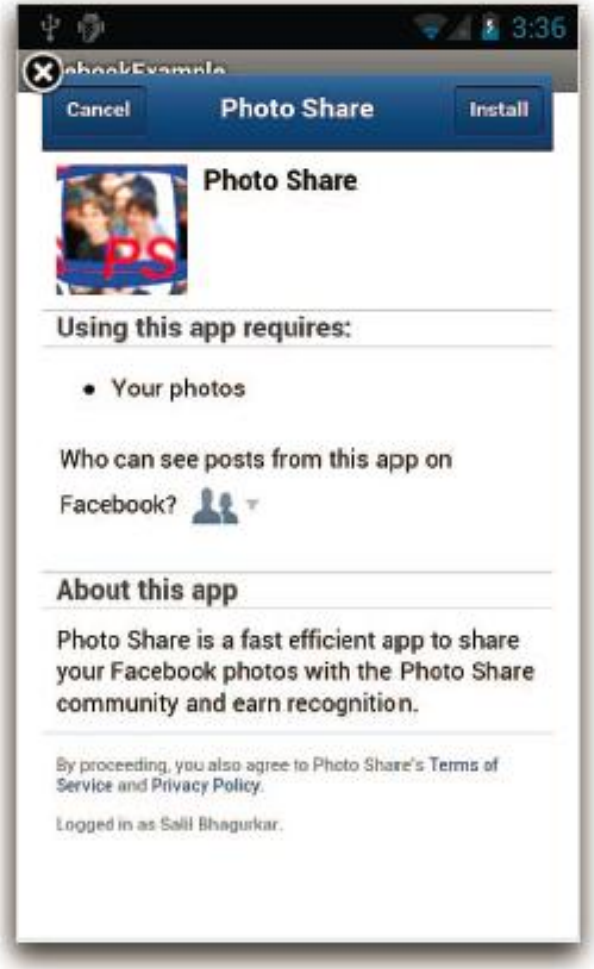
窃取用户登录信息（账号/密码）

```
myWebView.getSettings().setJavaScriptEnabled(true);
myWebView.addJavaScriptInterface(this , "JSInterface");
myWebView.loadUrl("javascript:" + contents of attack.js);

//JavaScript (attack.js)
var submitBtn = document.getElementById('btn_id');
submitBtn.onclick = function(){
  var email = document.getElementById('email_id').value;
  var password = document.getElementById('pwd_id').value;
  JSInterface.jsCall(email, password);
  return true;
}
```

修改授权页面


3:39
Cancel
Photo Share
Install
Photo Share
Using this app requires:
Your basic info
Your email address (msalil87@gmail.com)
Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
Your stories: checkins, events, notes, photos, status updates and videos
Who can see posts from this app on Facebook?


3:36
Cancel
Photo Share
Install
Photo Share
Using this app requires:
Your photos
Who can see posts from this app on Facebook?
About this app
Photo Share is a fast efficient app to share your Facebook photos with the Photo Share community and earn recognition.
By proceeding, you also agree to Photo Share's Terms of Service and Privacy Policy.
Logged in as Salil Bhagurkar.

国内主流SP支持的用户代理

Service Provider	User-agent
新浪微博	WebView/ SP app
腾讯微博/QQ/Qzone	WebView/ SP app/ Browser
微信	SP app
有道云笔记	WebView/ SP app
豆瓣	WebView
百度	WebView
人人网	WebView/ SP app
支付宝	WebView/ SP app

密钥/秘密管理

```
public OAuth() {
    super();
    this.oauth_consumer_key = "671dc036a4924fa39027abe4b7a7091a";
    this.oauth_consumer_secret = "22b430ced88be161924acdfe40dbb68c";
    this.oauth_signature_method = "HMAC-SHA1";
    this.oauth_timestamp = "";
```

```
// 注意！！此处必须设置appkey及appsecret，如何获取新浪微博appkey和appsecret请另外查询相关信息，此处不作介绍
private static final String CONSUMER_KEY = "------";// 替换为开发者的appkey，例如"1646212860";
private static final String CONSUMER_SECRET = "-------";// 替换为开发者的appkey，例如"94097772160b6f8ffc1315374d8861f9";
```

ANDROID平台中OAUTH实现的特点

密钥/秘密管理

```
root@X9077:/data/data/com.hupu.joggers/shared_prefs # ls -l
ls -l
-rw-rw---- u0_a428  u0_a428       305 2015-07-20 09:29 RONG_SDK.xml
-rw-rw---- u0_a428  u0_a428      1138 2015-07-20 09:24 cn_sharesdk_weibodb_Renre
n_2.xml
-rw-rw---- u0_a428  u0_a428       842 2015-07-20 09:28 cn_sharesdk_weibodb_SinaW
eibo_1.xml
-rw-rw-r-- u0_a428  u0_a428      1200 2015-07-20 09:28 hupu_mount.xml
-rw-rw-r-- u0_a428  u0_a428      6909 2015-07-20 09:36 hupurun.xml
-rw-rw---- u0_a428  u0_a428       204 2015-07-17 14:52 multidex.version.xml
-rw-rw---- u0_a428  u0_a428       375 2015-07-20 09:36 save_name.save_name.xml
-rw-rw---- u0_a428  u0_a428       148 2015-04-12 21:37 save_name.save_type.xml
-rw-rw---- u0_a428  u0_a428       502 2015-07-20 09:32 share_sdk_0.xml
-rw-rw-r-- u0_a428  u0_a428       302 2015-07-20 08:55 udp_new_file.xml
-rw-rw---- u0_a428  u0_a428       945 2015-07-20 09:32 umeng_general_config.xml
```

```
<int name="medal_best_ten_km" value="1000000000" />
<int name="user_is_new" value="1" />
<int name="medal_best_all_marathon" value="1000000000" />
<boolean name="isvisitor" value="false" />
<string name="renren">0</string>
<int name="distance_allbestma" value="0" />
<string name="distance_allbestdistance">4.15</string>
<int name="medal_best_time" value="2133" />
<string name="user_qqaccesstoken">56D010E4D1AE5308606A9E2(
```

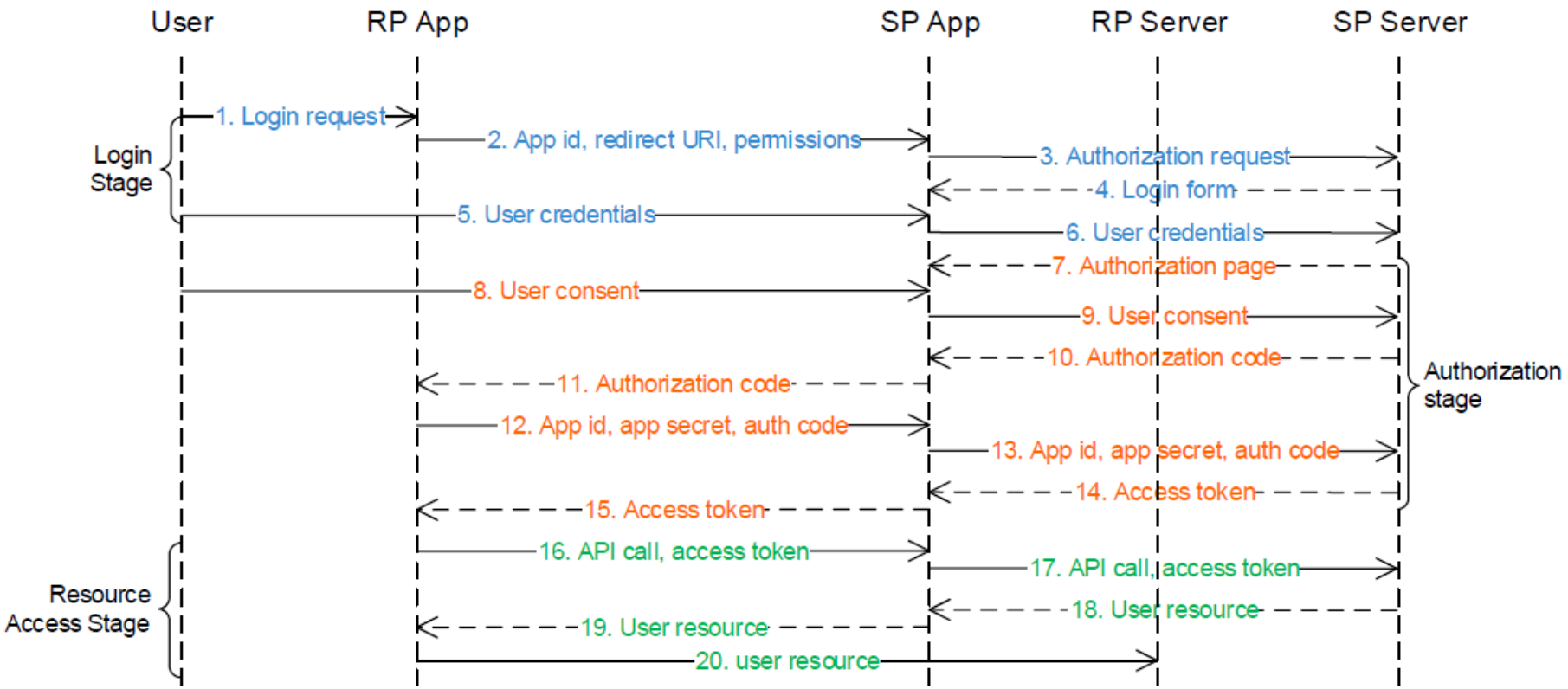
```
<string name="user_birthday">1991-7-11</string>
<int name="user_birth_day" value="11" />
<string name="weibo">0</string>
<int name="user_level" value="1" />
<string name="user_upgrade_rate">0.6</string>
<int name="medal_best_five_km" value="1000000000" />
<int name="distance_allbetten" value="0" />
<int name="user_numbyday" value="0" />
<int name="setting_sex" value="1" />
<boolean name="downdata" value="true" />
<int name="user_birth_year" value="1991" />
<string name="user_nextlevelvalue">20</string>
<int name="ads_is_click" value="1" />
<string name="weixin">0</string>
<string name="token">98290-1437356306-beb25d94e8e3224a96cd79c0e338671d</stri
g>
```

ANDROID平台中OAUTH实现的特点

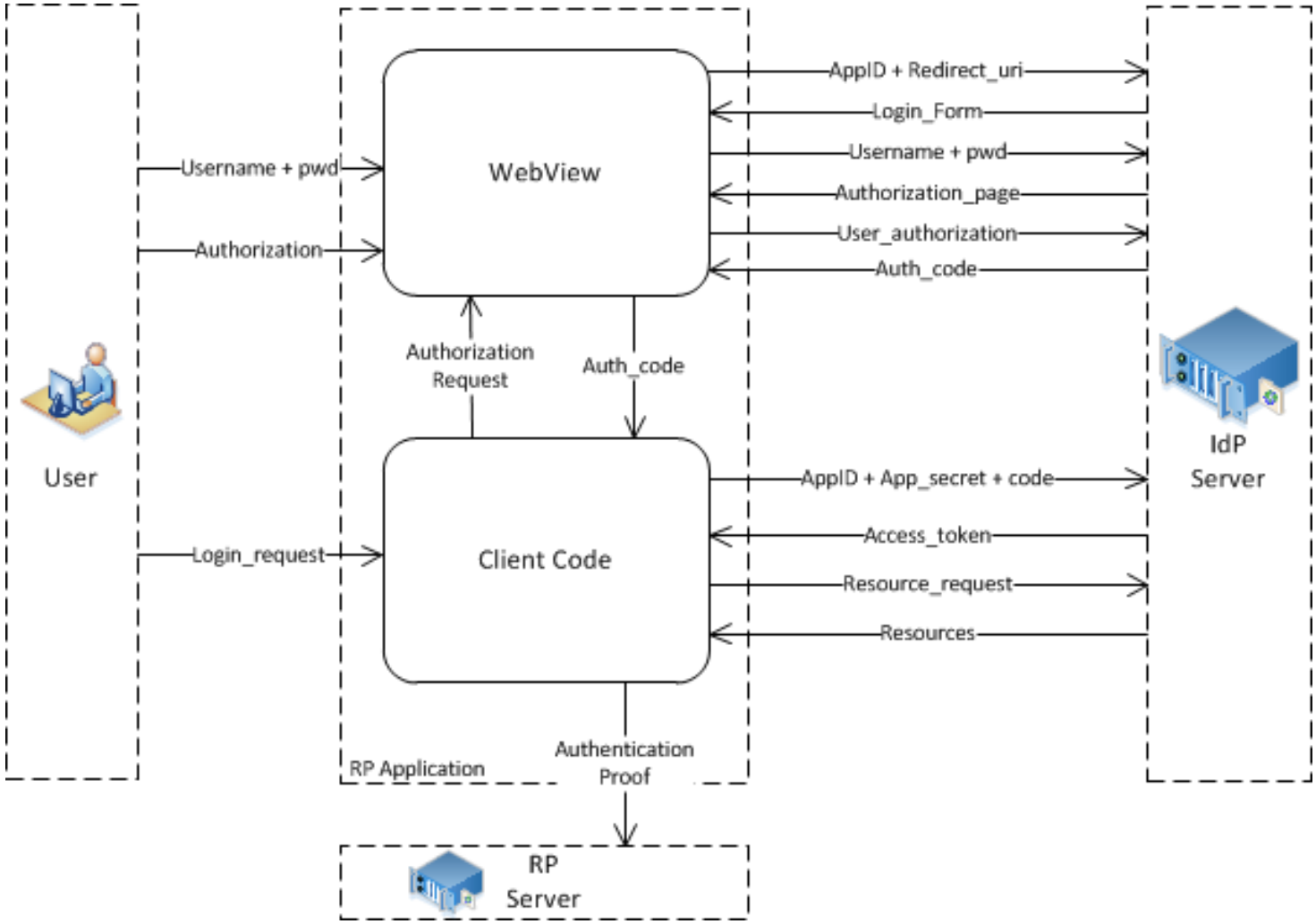
部分SP使用的OAUTH授权许可类型

Protocol	Service Providers
OAuth 1.0	Evernote、 Trello
OAuth 1.0a	Twitter、 Sina Weibo
OAuth 2.0 implicit grant	Sina Weibo、 Renren
OAuth 2.0 Modified implicit grant	Facebook、 Tencent
OAuth 2.0 Authorization code grant	Sina Weibo、 Sohu

安全审计模型-SP APP作为用户代理



安全审计模型-WEBVIEW作为用户代理



对ANDROID平台中OAUTH实现的安全审计

- 静态分析
- 流量分析

主流SP在OAUTH协议各阶段存在的安全问题

SP	Stage I		Stage II			Stage III
	V_1	V_3	V_1	V_2	V_3	V_5
Sina Weibo	√	×	√	×	×	√
Tencent Weibo	√	×	√	√	×	×
Qzone	√	×	√	√	×	×
QQ	√	×	√	√	×	×
Wechat	×	×	×	×	×	×
Youdao Note	√	√	√	×	√	×
Evernote	√	×	√	√	×	×
Yixin	√	×	√	×	×	×
Douban	√	×	√	√	×	×
Renren	√	√	√	√	√	√
Kaixin	√	√	√	√	√	×
Baidu	√	×	√	√	×	×
Taobao	√	√	√	√	√	×
Laiwang	×	×	×	√	×	×
Alipay	√	√	√	√	√	×

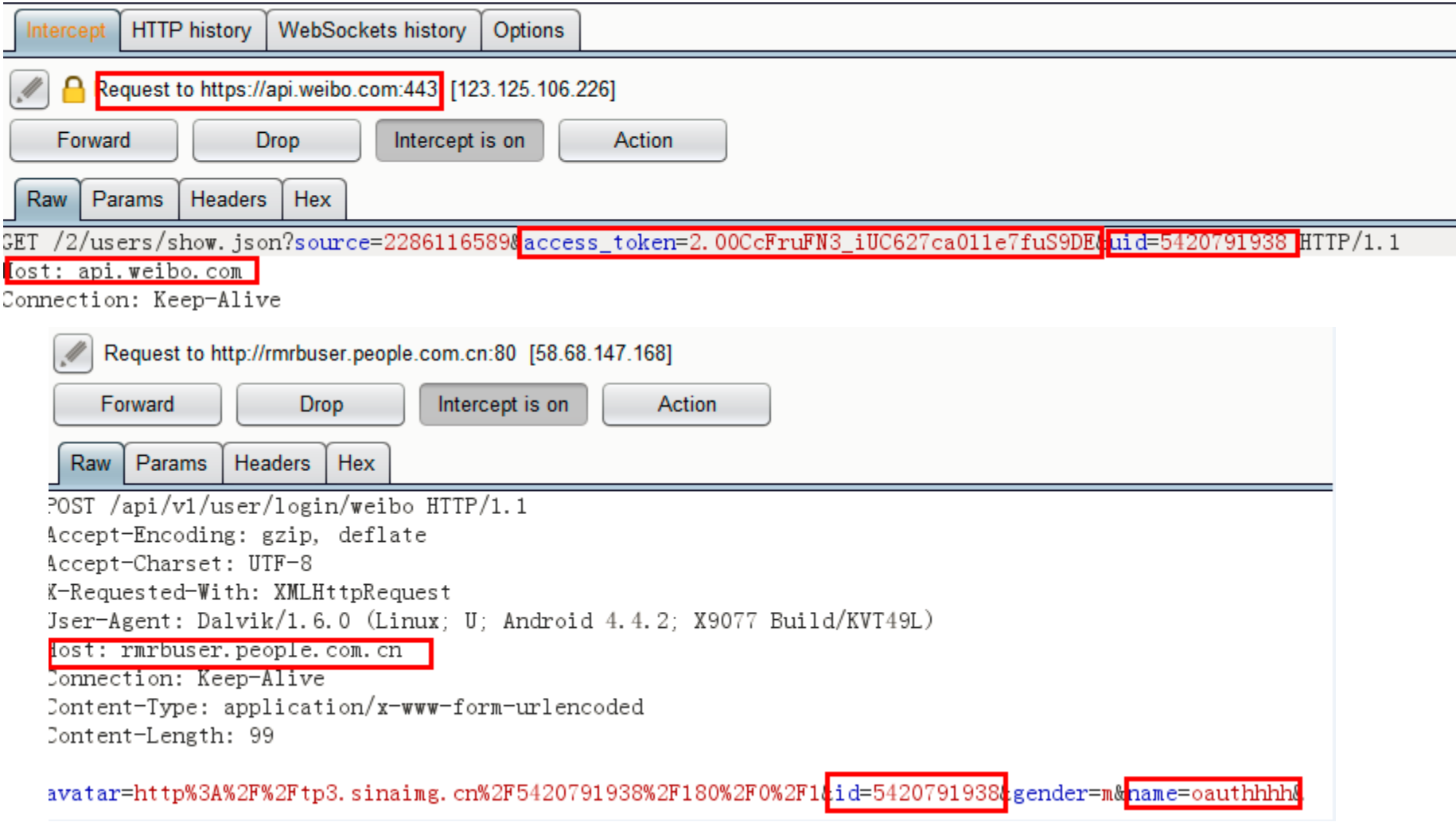
主要漏洞类型

- 不安全的用户代理（V1）
- 缺乏协议参与者身份认证（V2）
- 不安全的信息传输（V3）
- 不安全的秘密管理（V4）
- 不正确的服务器端参数校验（V5）
- 不正确的认证凭据（V6）

案例——人民日报 & 新浪微博


中国移动3G
15:43
人民日报客户端
请输入手机号/E-mail
请输入密码
登录
注册
忘记密码
用户协议
还可用以下账号登录
人民网通行证
新浪微博
QQ


中国移动3G
15:51
新浪微博
Powered by ShareSDK
微博
weibo.com
注册
授权 人民日报客户端 访问你的微博帐号
请用微博帐号登录
请输入密码
登录


Intercept
HTTP history
WebSockets history
Options
Request to https://api.weibo.com:443 [123.125.106.226]
Forward
Drop
Intercept is on
Action
Raw
Params
Headers
Hex
GET /2/users/show.json?source=2286116589&access_token=2.00CcFruFN3_iUC627ca011e7fuS9DE&uid=5420791938 HTTP/1.1
Host: api.weibo.com
Connection: Keep-Alive
Request to http://rmrbuser.people.com.cn:80 [58.68.147.168]
Forward
Drop
Intercept is on
Action
Raw
Params
Headers
Hex
POST /api/v1/user/login/weibo HTTP/1.1
Accept-Encoding: gzip, deflate
Accept-Charset: UTF-8
X-Requested-With: XMLHttpRequest
User-Agent: Dalvik/1.6.0 (Linux; U; Android 4.4.2; X9077 Build/KVT49L)
Host: rmrbuser.people.com.cn
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 99
avatar=http%3A%2F%2Ftp3.sinaimg.cn%2F5420791938%2F180%2F0%2F1&id=5420791938&gender=m&name=oauthhhh&

Intercept | HTTP history | WebSockets history | Options

Request to https://api.weibo.com:443 [123.125.106.226]

Forward | Drop | Intercept is on | Action

Raw | Params | Headers | Hex

```
GET /2/users/show.json?source=2286116589&access_token=2.00fNy4UDN3_iUCeddc58ed6elsoDyB&uid=3197401607 HTTP/1.1
Host: api.weibo.com
Connection: Keep-Alive
```

Thank you !

Thank you !

Thank you !

Thank you !