



EISS-2019企业信息安全峰会

北京站/3.29





业务安全红蓝对抗的探索与实践

柳兮 2019.3.29



归零实验室-简介

- 柳兮-阿里安全归零实验室
- 归零实验室简介

阿里安全归零实验室成立于2017年11月，实验室致力于对黑灰产技术的研究,愿景通过技术手段解决当前日益严重的网络违规和网络犯罪问题，为阿里新经济体保驾护航。

目前团队也在不断的招聘各种优秀人才，研发专家、数据分析专家、情报分析与体系化专家等，欢迎加盟，联系邮箱
back2zero@service.alibaba.com



阿里安全

备注说明

**阅读PPT前，建议先阅读FIT 2019
议题《如何做好业务安全红蓝对抗》**

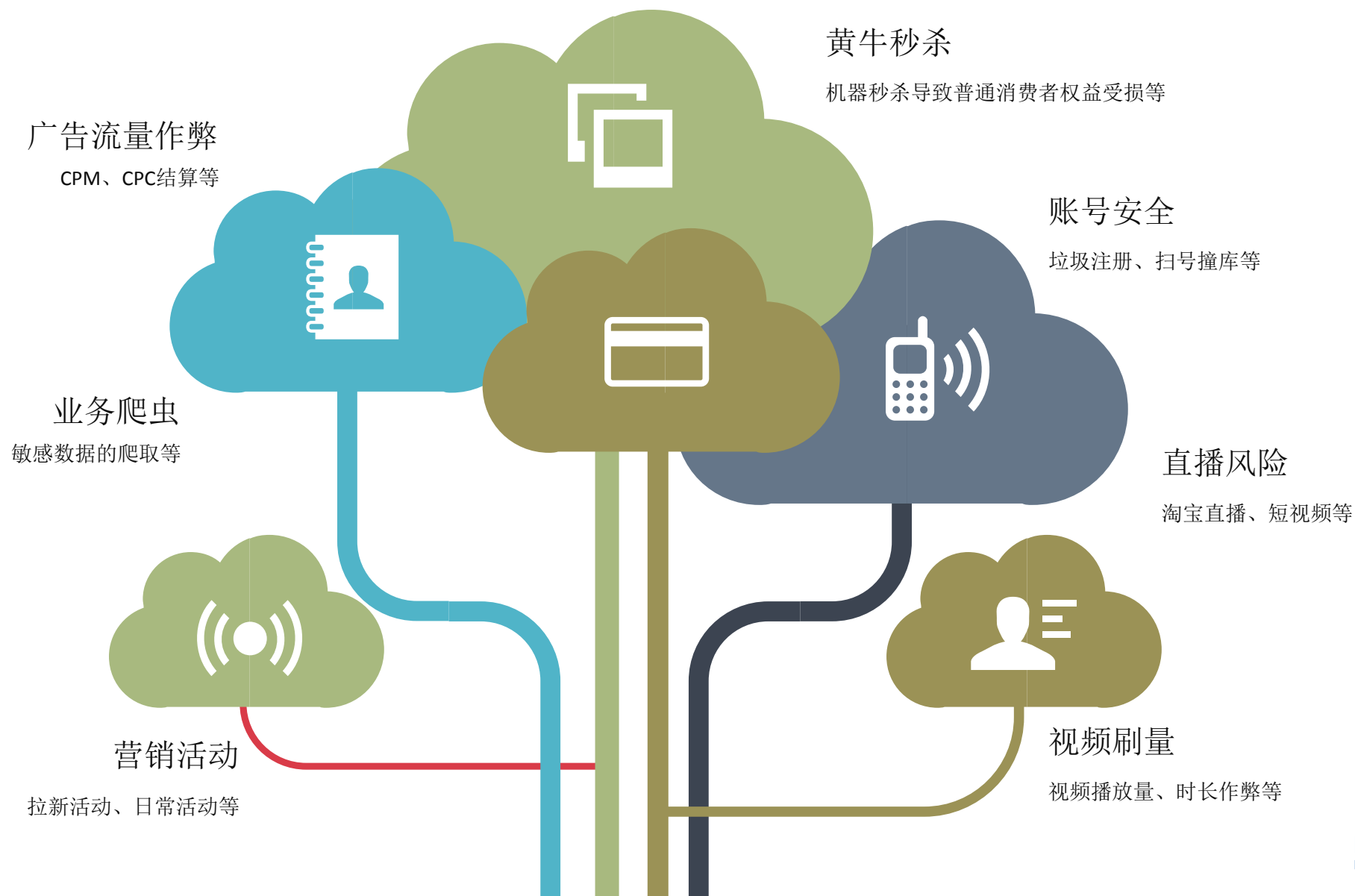
主要内容

- 常见的业务风险和风控体系
- 业务红蓝对抗的探索实践过程
- 砺剑蓝军演练平台

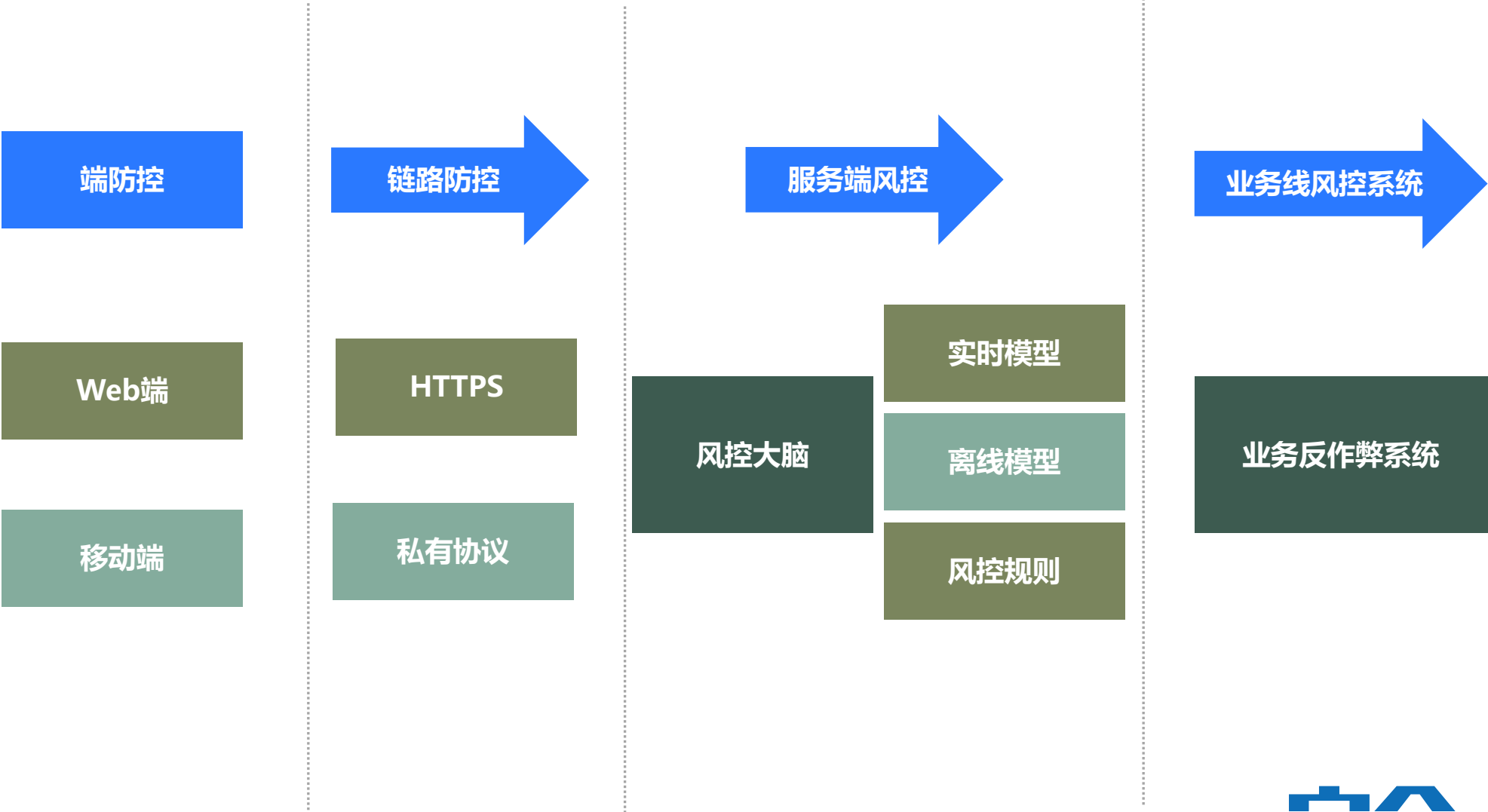
01 / 常见的业务风险和风控体系



常见的业务风险



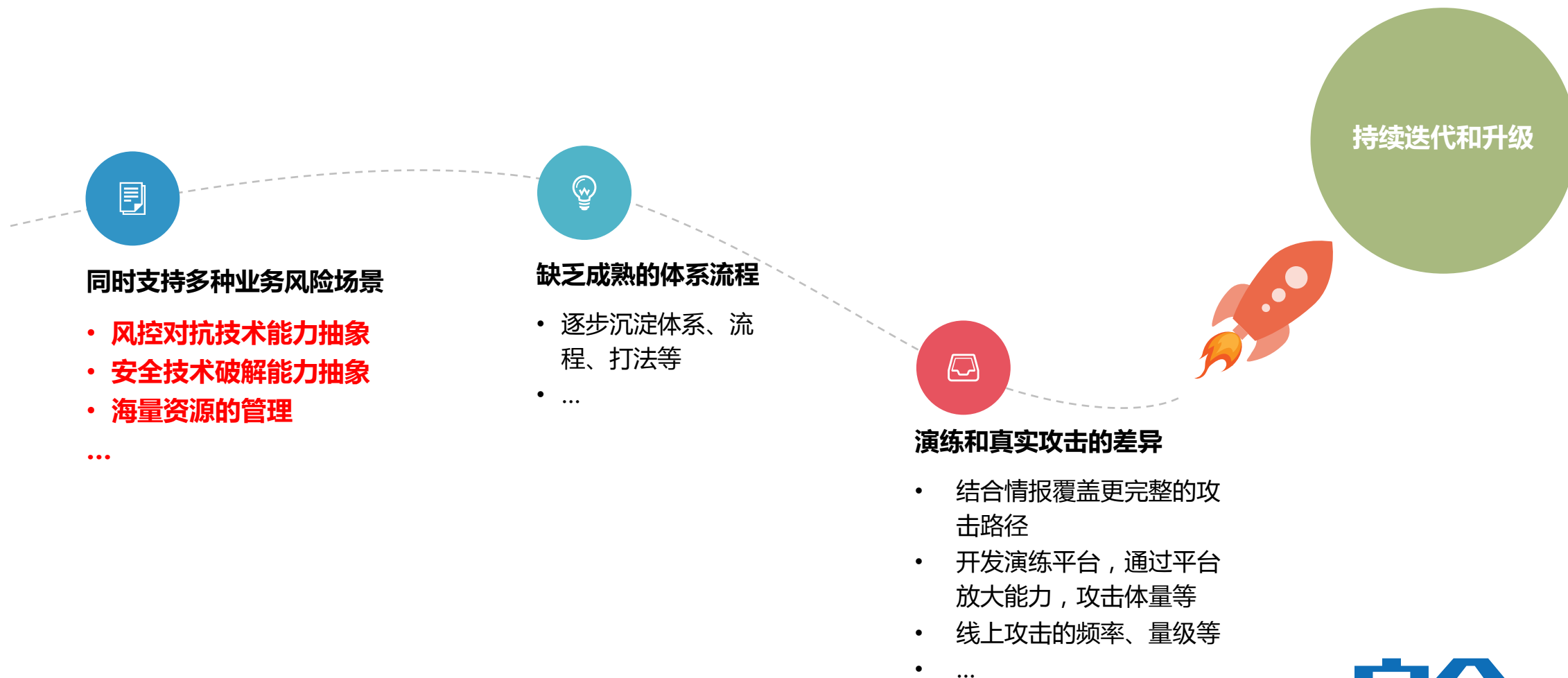
常见的风控体系



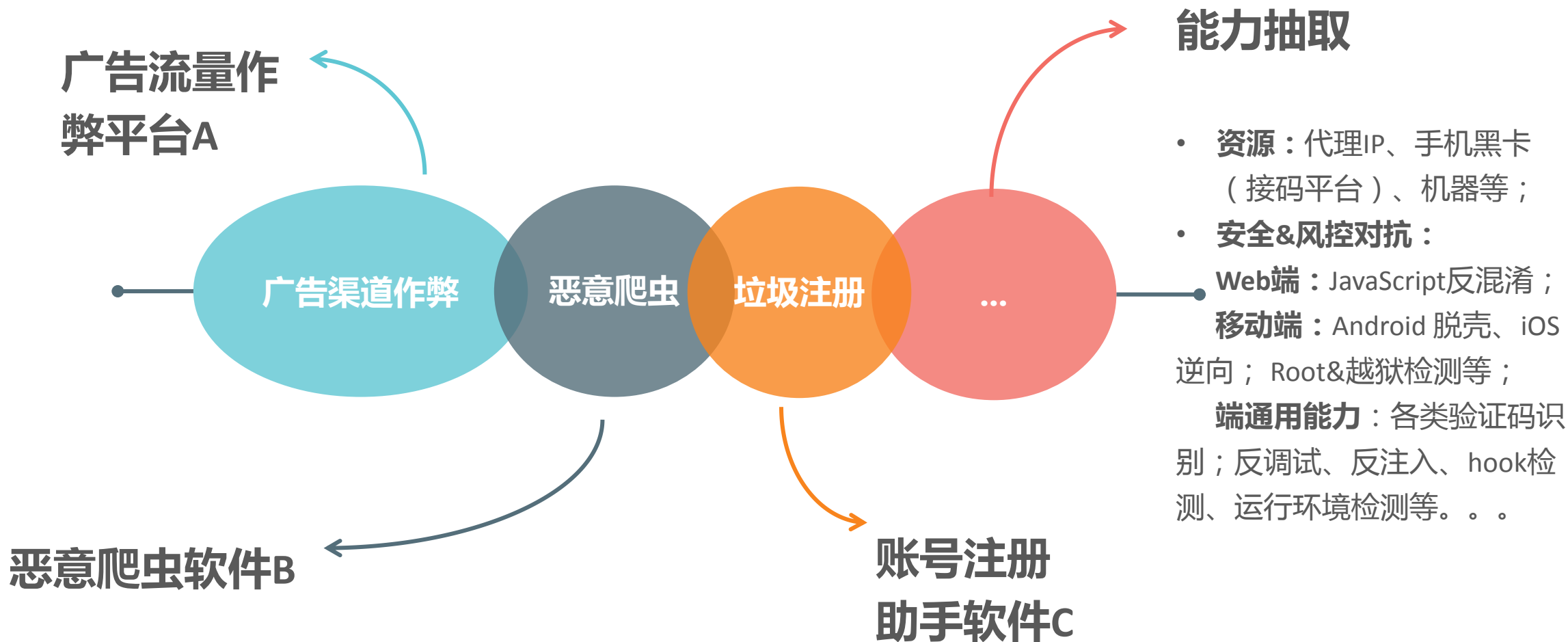
02 / 业务红蓝对抗的探索实践过程

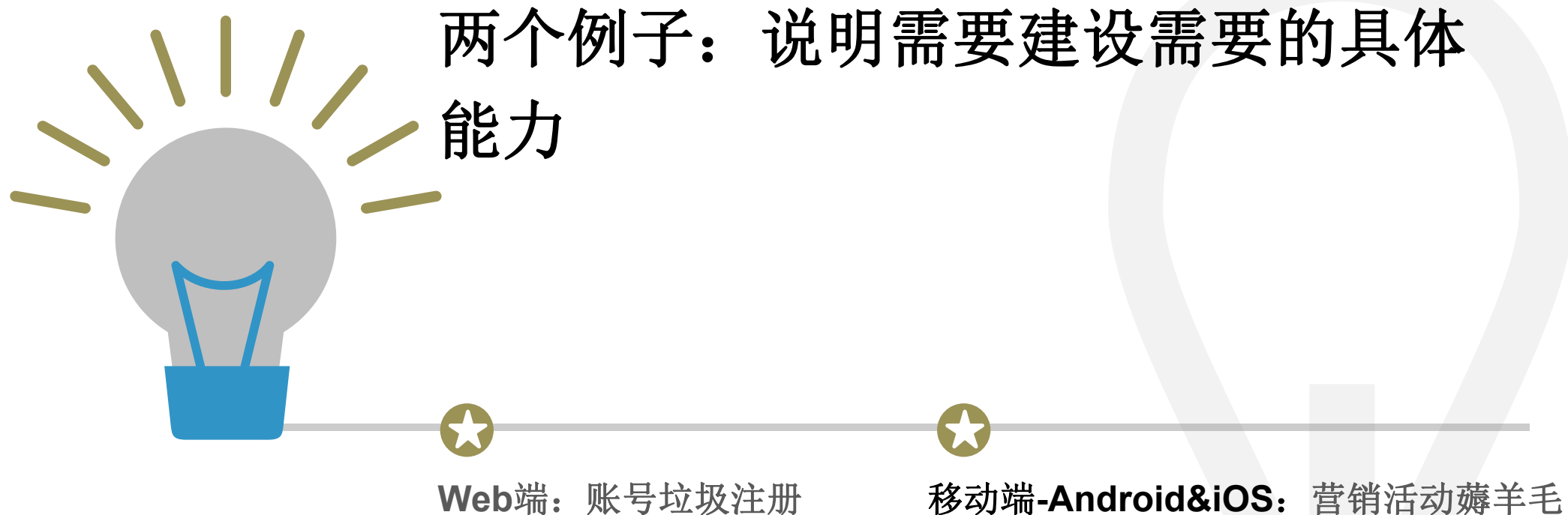


难点和解法



探索实践过程-业务风险场景能力抽象





探索实践过程-Web端垃圾注册-资源

序号	账号	登陆密码	支付密码	状态

分隔符: ---- 数据之间的区分符号 删除选中 全部删除 保存导出 全选 反选 全消

1W以上大数据或没有显卡的电脑, 不建议开启显示和跟踪, 更节省资源。 ☒ 开启列表提示 ☐ 开启数据跟踪

宽带账号: ☐ 开启拨号 拨号控制: 50

宽带密码: ADSL拨号模式: 断开等待(秒): 15 ☐ 过滤重复IP 连接 断开

检测联网模式: 系统联网状态 每台电脑的要求和系统稳定性不同, 用户可测试来选择稳定的拨号模式。
拨号完, 软件需要检测是否真正的连上网络, 这里提供3种模式, 足以应付所有的电脑和服务。【非必要选择】

平台账号: 平台密码: 地区: 登陆平台

账号: 密码: 题分: 登陆

线程数量: 1 任务间隔(秒): 0.5 线程超时(秒): 300 指定数停止: 100 开始任务

账号: @ (3-4), # (4-7) 邮箱后缀: @.com 登陆密码: qweqwe123 支付密码: 123234

#数字 @字母 *字母或数字 %汉字 \$汉字拼音 %姓名 %Y年 %M月 %D日 %H时 %I分 %S秒 【类型随意组合, 长度可控制: # (1-5) @ (2-5), 各种符号代表各种类别, 括号内的数字代表随机的位数, 括号是小的, 不随机的固定, 也可

黑灰产垃圾注册软件：
使用猫池、接码平台



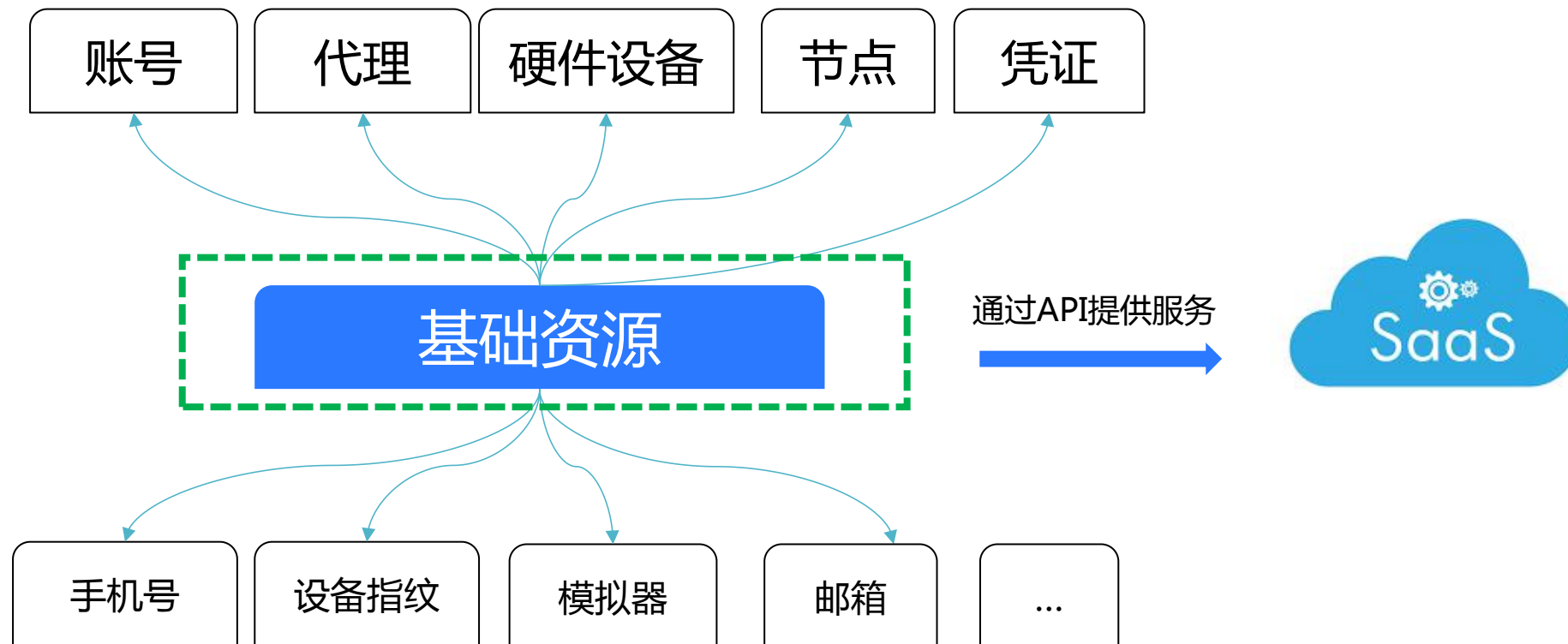
黑灰产垃圾注册软件：
使用代理

**HTTP代理

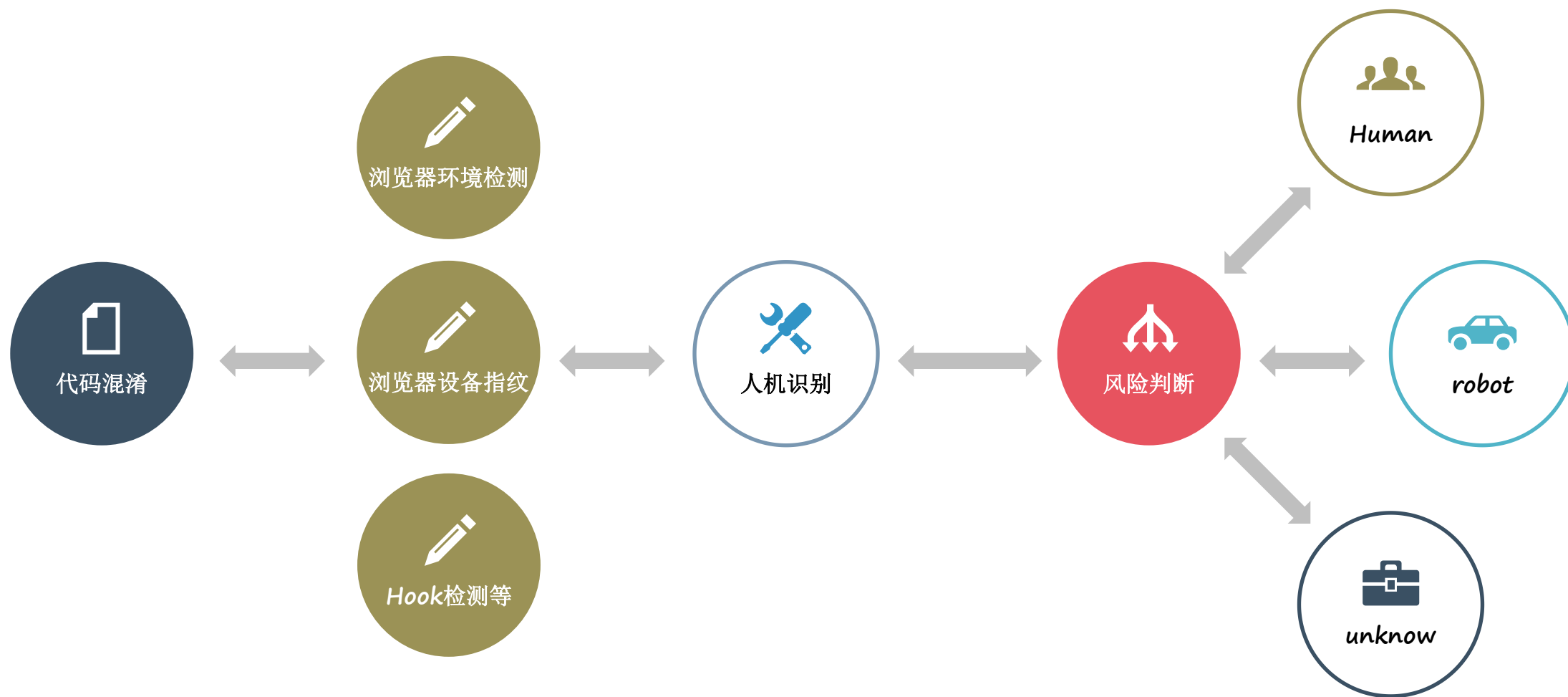
您首选的代理IP服务供应商



探索实践过程-Web端垃圾注册-资源管理

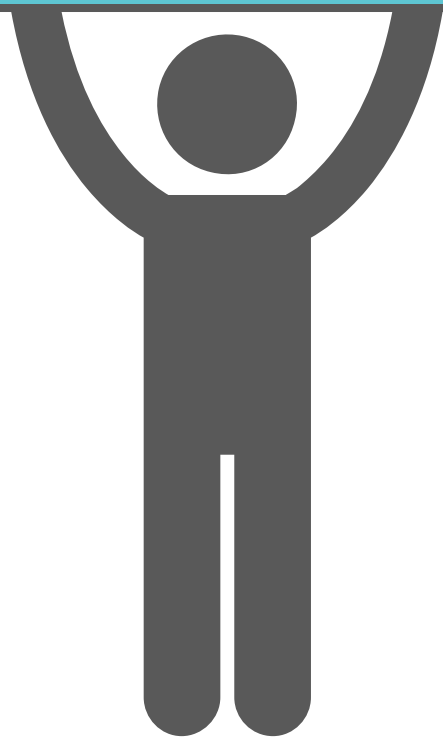


探索实践过程-Web端垃圾注册-安全&风控对抗



探索实践过程-Web端垃圾注册-安全&风控对抗

前端代码混淆对抗



01

非常简单的脚本压缩

例如uglify等，并不是真正的混淆，很好还原

复杂的脚本编码

例如jjencode、aaencode、jsfuck等编码，也很好还原

02

03

修改语法树进行混淆

代码执行流程没有改变，耐心分析也可以还原

变更控制流进行混淆

打乱原有代码流程，插入逻辑上无关的代码，再保证混淆的更新频率，理论上相对安全

04

探索实践过程-Web端垃圾注册-JavaScript混淆对抗

jjencode demo

Be aware

Using `jjencode` for actual attack isn't good idea.

- Decode easily. `jjencode` is not utilitarian obfuscation, just an encoder.
- Too characteristic. Detected easily.
- Browser depended. The code can't run on some kind of browsers.

Enter any JavaScript source:

```
alert("Hello, EISS" )
```

global variable name used by jjencode : ☐ palindrome

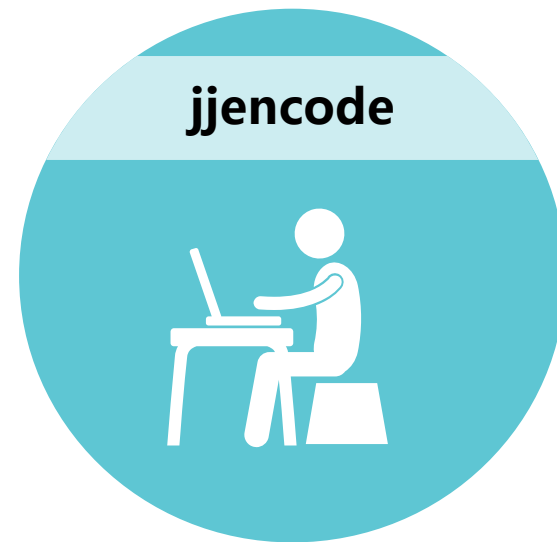
[illegible]

Input the JJEncode here:

[illegible]

Decoded JJEncode String:

```
alert("Hello, EISS" )
```



探索实践过程-Web端垃圾注册-JavaScript混淆对抗

通过动态调试，结合hook、snippets等方式来分析还原代码

aaencode demo

aaencode - Encode any JavaScript program to Japanese style emoticons (^_^)

Enter JavaScript source:

```
alert("Hello, EISS")
```

[illegible]

```

0 o = {o: (-)};
1 c = {c: (-)};
2 d = {d: (-)};
3 e = {e: (-)};
4 f = {f: (-)};
5 g = {g: (-)};
6 h = {h: (-)};
7 i = {i: (-)};
8 j = {j: (-)};
9 k = {k: (-)};
10 l = {l: (-)};
11 m = {m: (-)};
12 n = {n: (-)};
13 o = {o: (-)};
14 p = {p: (-)};
15 q = {q: (-)};
16 r = {r: (-)};
17 s = {s: (-)};
18 t = {t: (-)};
19 u = {u: (-)};
20 v = {v: (-)};
21 w = {w: (-)};
22 x = {x: (-)};
23 y = {y: (-)};
24 z = {z: (-)};

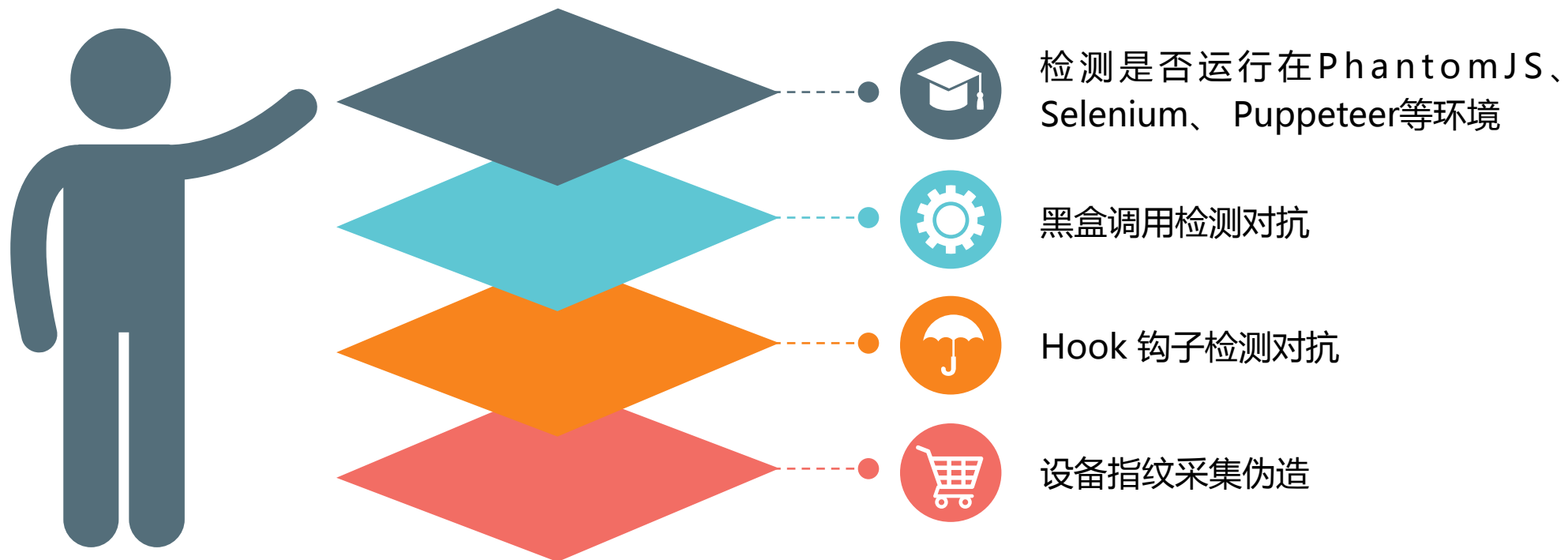
```

探索：通过Partial Evaluation实现的JavaScript反混淆

```
1 !function(){function e(e,a){function c(e){for(var a=1;void 0!==(a);){var r=7&a,b=
2 case 16:fs=M?7618:14400;break;case 17:le=re[xe],fs=le?1732:3682;break;case 18:Q
3 break;case 4:he=Ce.charCodeAt($)-106,ee+=String.fromCharCode(he),fs=1512;break;
4 case 0:fs=U?18565:21926;break;case 1:H=127&U,U>==7,fs=U?9667:20199;break;case 2
5 fs=d?232:10407;break;case 22:$++,fs=12673;break;case 23:v=T,T=v,E=E.concat(T),f
6 Z="","",_=0,fs=20163;break;case 17:L=T,T=L<<11,v|=T,E.push(5),T=void 0,L=v,v=[],M=
7 Es[Y](Rs),Y="co",Y+="nec",Y+="t",fs=19968;break;case 5:Xe=le[Be],Ee=G-1,Xe+=Ee
8 v++,fs=24900;break;case 14:T=M,M=T<<4,v|=M,T=C[L],fs=T?22150:22598;break;case 1
```

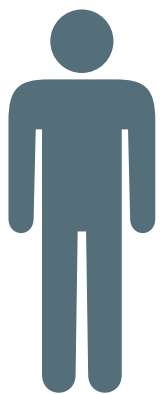


探索实践过程-Web端垃圾注册-代码运行环境检测对抗



探索实践过程-Web端垃圾注册-代码运行环境检测对抗

环境检测



BOM、DOM等 特征差异

PhantomJS、Chrome Headless : window.callPhantom、window.phantom、User-Agent、navigator.plugins.length === 0、navigator.languages == ''、window.outerWidth === 0 || window.outerHeight === 0 等

Selenium : window.domAutomation、window.domAutomationController、window.webdriver、__driver_evaluate、__webdriver_evaluate、__selenium_evaluate、__fxdriver_evaluate、__driver_unwrapped、__webdriver_unwrapped、__selenium_unwrapped、__fxdriver_unwrapped等

Callstack追踪、HTML5新特性等

WebAudio、WebGL、Canvas、WebSocket等

Some tricks

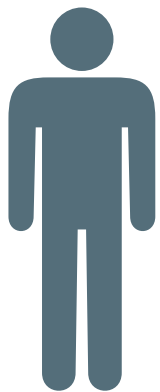
unicode特殊编码过的代码（部分Chrome版本） ☺

```
> var a = '□';  
console.log("test");  
  
> var a = '□';  
console.log("test");  
  
> console.log("test--1");  
test--1
```



探索实践过程-Web端垃圾注册-Hook检测对抗

钩子检测



一个简单的例子：alert()

```
> 19:14:48.429 alert.toString()
< 19:14:48.432 "function alert() { [native code] }"

> 19:15:06.020 var oldAlert = alert;
< 19:15:06.024 undefined

> 19:16:04.481 alert = function(m) {
    console.log("fake alert...")
    oldAlert(m);
};
< 19:16:04.488 f (m) {
    console.log("fake alert...")
    oldAlert(m);
}

> 19:16:13.551 alert.toString()
< 19:16:13.554 "function(m) {
    console.log("fake alert...")
    oldAlert(m);
}"

> 17:54:07.970 alert.toString = function() {
    return 'function alert() { [native code] }';
};
< 17:54:07.975 f () {
    return 'function alert() { [native code] }';
}

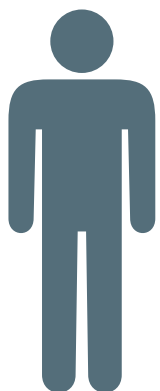
> 17:54:09.993 alert.toString()
< 17:54:09.997 "function alert() { [native code] }"
```

调用原型上的Function.prototype.toString.call(alert) 检测，
继续hook ... 持续攻防迭代 ☺



探索实践过程-Web端垃圾注册-设备指纹伪造

设备指纹



多维度特征采集

开源的fingerprint2.js (效果并不是很好, 只是拿来举个例子 😊)

```
395 var userAgent = function (done) {
396   done(navigator.userAgent)
397 }
398 var webdriver = function (done, options) {
399   done(navigator.webdriver === null ? options.NOT_AVAILABLE : navigator.webdriver)
400 }
401 var languageKey = function (done, options) {
402   done(navigator.language || navigator.userLanguage || navigator.browserLanguage || navigator.systemLanguage || options.NOT_AVAILABLE)
403 }
404 var colorDepthKey = function (done, options) {
405   done(window.screen.colorDepth || options.NOT_AVAILABLE)
406 }
407 var deviceMemoryKey = function (done, options) {
408   done(navigator.deviceMemory || options.NOT_AVAILABLE)
409 }
410 var pixelRatioKey = function (done, options) {
411   done(window.devicePixelRatio || options.NOT_AVAILABLE)
412 }
413 var screenResolutionKey = function (done, options) {
414   done(getScreenResolution(options))
415 }
416 var getScreenResolution = function (options) {
```

反混淆js分析后, 进行伪造即可... evercookie方案同理, 不展开😊

```
> 19:29:53.601 navigator.platform
< 19:29:53.605 "MacIntel"
> 19:30:27.690 // example
Object.defineProperty(navigator, "platform", {
  get: function() {
    return "iPhone";
  }
});
< 19:30:27.693 ▶ Navigator {vendorSub: "", productSub: "20030107", vendor: "Google Inc.", n
> 19:30:30.392 navigator.platform
< 19:30:30.395 "iPhone"
> |
```



探索实践过程-Web端垃圾注册-人机识别对抗



人机识别攻防

图灵测试-验证码

Deep Learning，通过卷积神经网络CNN识别等

用户行为

伪造鼠标轨迹、行为事件、键盘事件等

大数据模型

设备牧场，养号对抗等

探索实践过程-Web端垃圾注册-风控算法模型&规则对抗



01

高质量代理IP

对抗IP画像☺

02

算法模型

基于Logistic 等算法模型的对抗☺

03

场景特有的规则

注册频率、UA分布比例等☺

探索实践过程-移动端 营销活动薅羊毛

时间原因不展开说，思路同Web端 😊

设备伪造

Android、iOS一键改机软件，模拟器检测等

模拟定位

随机参数

终端伪装

新机参数

iPhone 5S/WIFI/9.3.2/剩:9.60 GB

应用列表[1]/

保存参数

全息备份

一键新机

清理Safari

备份记录

清理剪贴板

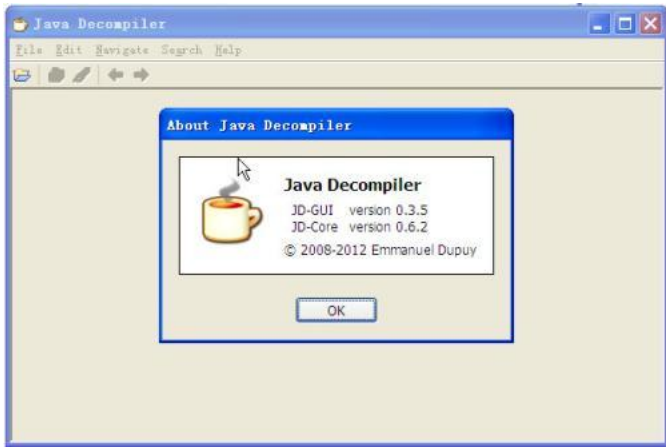
清理Keychain

原始机器

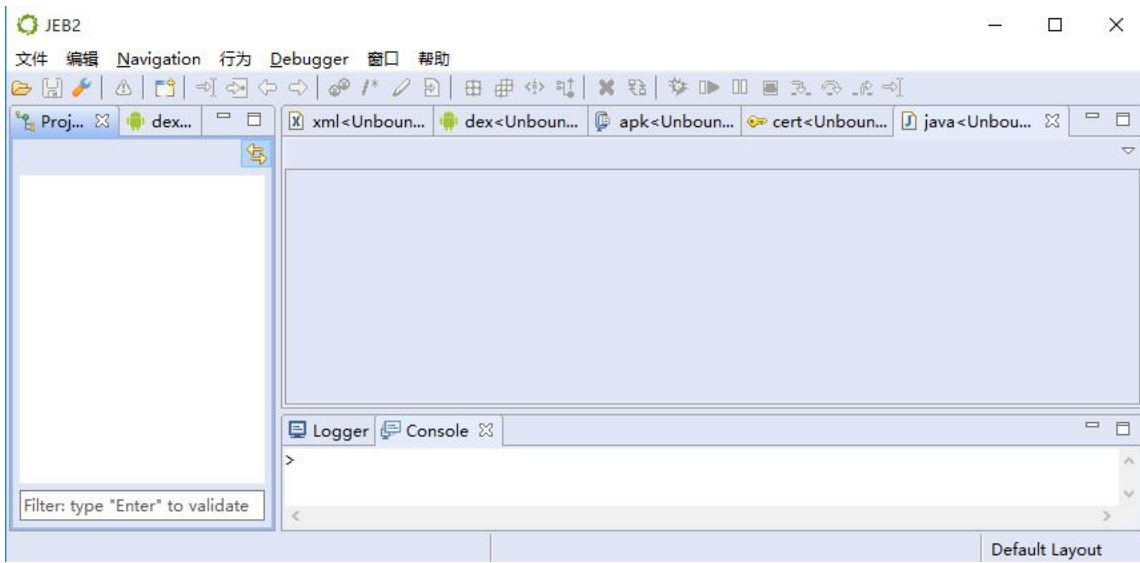
只新机不备份则不打开

协议破解

刷接口API、Hook等



FRIDA



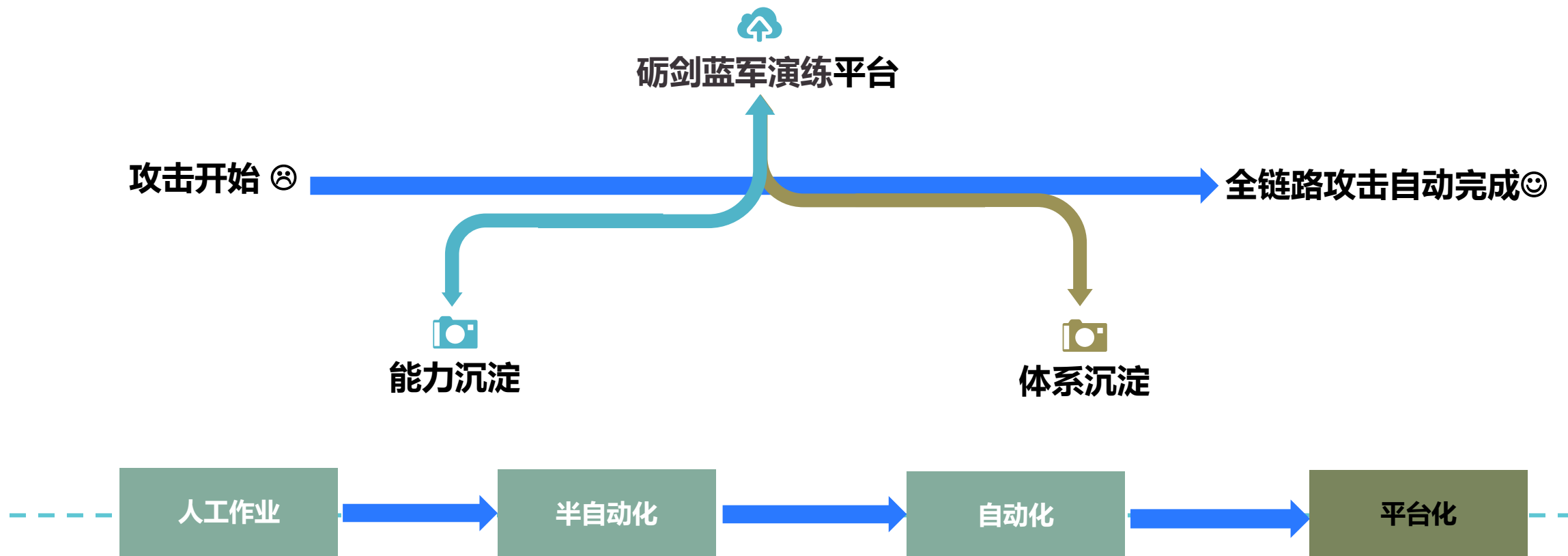
```
iPod:~ root# DYLD_INSERT_LIBRARIES=dumpdecrypted.dylib /var/mobile/Applications/xxxxxxx-xxxx-x  
mach-o decryption dumper
```

DISCLAIMER: This tool is only meant for security research purposes, not for application cracker

```
[+] Found encrypted data at address 00002000 of length 1826816 bytes - type 1.  
[+] Opening /private/var/mobile/Applications/xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/Scan.app/Scan  
[+] Reading header  
[+] Detecting header type  
[+] Executable is a FAT image - searching for right architecture  
[+] Correct arch is at offset 2408224 in the file  
[+] Opening Scan.decrvpted for writing.
```



探索实践总结



03 / 砺剑蓝军演练平台

砺剑蓝军演练平台




Q&A

招人：

- 1.移动安全逆向工程师&专家（Android&iOS）
- 2.红蓝对抗攻防演练工程师（web&移动应用安全、渗透测试、业务风控安全技术研究、漏洞挖掘等方向）

工作地点：北京、杭州 皆可



柳兮 

中国



扫一扫上面的二维码图案，加我微信

Thank You

