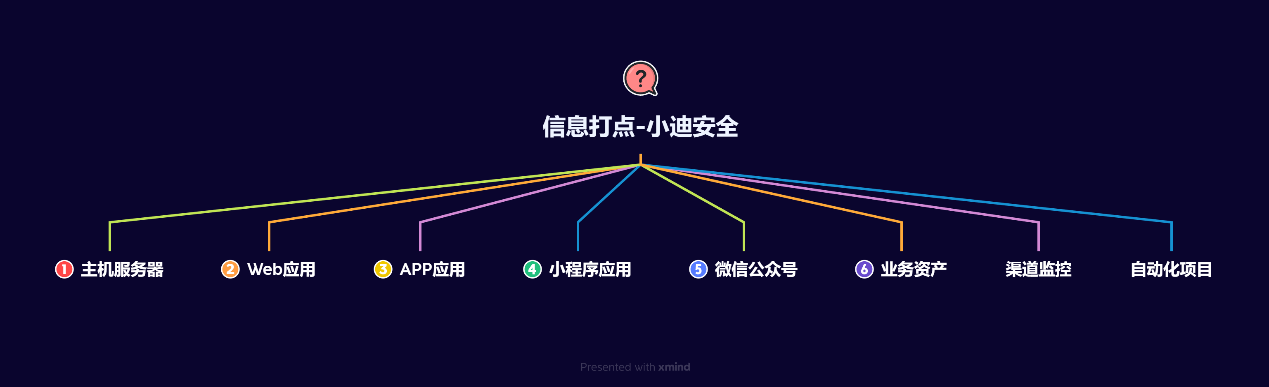
信息收集-小程序应用&逆向反编译&外在抓包&动态调试&自动化提取&主包分包源码



#知识点：  
1、信息收集-服务器系统-操作系统&IP资产  
2、信息收集-服务器系统-端口扫描&服务定性  
---------------------------------------------------  
1、信息收集-Web应用-机构产权&域名相关性  
2、信息收集-Web应用-DNS&证书&枚举子域名  
---------------------------------------------------  
1、信息收集-Web应用-架构分析&指纹识别  
2、信息收集-Web应用-架构分析&WAF&蜜罐  
3、信息收集-Web应用-架构分析&框架组件识别  
---------------------------------------------------  
1、信息收集-Web应用-源码获取-已知指纹&未知指纹  
2、信息收集-Web应用-源码获取-泄漏问题&发现指纹  
---------------------------------------------------  
1、信息收集-Web应用-JS提取分析-人工&插件\*项目  
2、信息收集-Web应用-JS提取分析-URL&配置&逻辑  
---------------------------------------------------  
1、信息收集-Web应用-CDN加速-如何识别  
2、信息收集-Web应用-CDN加速-绕过方法  
---------------------------------------------------  
1、信息收集-Web应用-开发框架-识别安全  
2、信息收集-Web应用-安全组件-特征分析  
---------------------------------------------------  
1、信息收集-APP应用-公开信息-知识产权&开发者定位  
2、信息收集-APP应用-资产信息-抓包&静态提取&动态调试  
---------------------------------------------------  
1、信息收集-小程序应用-公开信息-知识产权&平台搜索  
2、信息收集-小程序应用-资产信息-抓包&提取&动态调试  
  
#章节点：(待补充)  
#具体点：(待补充)

演示案例：

* 小程序应用-公开信息-知识产权&平台搜索
* 小程序应用-资产信息-抓包&提取&动态调试

APP渗透测试的范围应涵盖APP所有功能和组件，包括但不限于以下几个方面：  
• 前端安全：包括界面交互、输入验证、会话管理等。  
• 后端安全：包括API接口、数据库交互、业务逻辑等。  
• 数据传输安全：包括数据传输过程中的加密、签名、完整性校验等。  
• 用户认证与授权：包括登录、注册、权限分配、会话管理等。  
• 第三方库与组件：包括使用的第三方库、SDK、插件等的安全性和稳定性。  
• 配置与环境：包括APP的配置文件、环境变量、日志管理等。  
  
移动安全：  
1、站着逆向角度  
有无保护，源码修改打包HOOK  
删除验证代码 证书 代理 抓包  
删除验证代码 破解版（突破会员制度等）  
翻到密钥 接口 密码等关键敏感  
2、站着渗透角度  
APP架构资产-表现，源码，调试逻辑  
web 域名 cms api接口 web安全问题  
ip 端口 服务 web服务（见上） 数据库 中间件 等服务测试  
API 接口 测试   
逻辑安全问题  
  
通过获取App配置、数据包，去获取url、api、osskey、js等敏感信息。  
1、资产信息-IP 域名 网站 -转到对应Web测试 API接口测试 服务测试   
2、泄露信息-配置key 资源文件 - key（osskey利用，密钥配置等）  
3、代码信息-代码安全 HOOK绕过限制（证书，代理，脱壳等）- 逆向相关  
APP中收集资产：通讯抓包，静态提取，动态调试  
  
#公开信息收集：  
通过搜索引擎、社交媒体、应用商店等渠道，收集目标APP信息，如APP名称、版本、开发者、下载量等。名称获取APP信息（爱企查/小蓝本/七麦/点点）  
1、查备案信息在搜  
2、网站上有APP下载  
3、市场直接搜单位名称  
https://www.qimai.cn/  
https://app.diandian.com/  
https://aiqicha.baidu.com/  
https://www.xiaolanben.com/  
  
#APP资产提取：  
1、动态抓包  
Burpsuite+Reqable  
参考前期部分抓包技术  
技术优点：没有误报  
技术缺点：无法做到完整  
  
2、静态提取-反编译  
技术优点：数据较为完整  
技术缺点：有很多无用的资产  
APKDeepLens收集APP的权限、组件信息、IP、Secret。  
https://github.com/d78ui98/APKDeepLens  
  
APKLeaks工具，用于检查APK文件中的硬编码URL、API密钥等敏感信息。  
https://github.com/dwisiswant0/apkleaks  
  
AppInfoScanner对Android、iOS、WEB多场景应用进行信息扫描，收集IP、URL、Server、CDN等信息。  
https://github.com/kelvinBen/AppInfoScanner  
  
3、动态调试-反调试  
技术优点：HOOK可解决不能抓包不能代理等情况  
技术优点；搞逆向的人能看到实时的app调用链等  
技术缺点：部分APP有反调试等无法做到完整  
MobSF是一种自动化的移动应用（Android/iOS/Windows）静态和动态分析的测试，恶意软件分析和安全评估框架。  
https://github.com/MobSF/Mobile-Security-Framework-MobSF

#小程序获取-各大平台&关键字搜索  
-微信  
-百度  
-支付宝  
-抖音头条  
  
#微信小程序-源码架构解析  
1.主体结构  
小程序包含一个描述整体程序的 app 和多个描述各自页面的 page。  
一个小程序主体部分(即app)由三个文件组成，必须放在项目的根目录，如下：  
 文件 必需 作用   
 app.js 是 小程序逻辑  
 app.json 是 小程序公共配置  
 app.wxss 否 小程序公共样式表  
2.一个小程序页面由四个文件组成，分别是:   
 xxx.js 页面逻辑  
 xxx.json 页面配置  
 xxx.wxml 页面结构  
 xxx.wxss 页面样式  
3.项目整体目录结构  
 pages 页面文件夹  
 index 首页  
 logs 日志  
 utils   
 util 工具类(mina框架自动生成,你也可建立：api)  
 app.js 入口js(类似于java类中的main方法)、全局js  
 app.json 全局配置文件  
 app.wxss 全局样式文件  
 project.config.json 跟你在详情中勾选的配置一样  
 sitemap.json 用来配置小程序及其页面是否允许被微信索引  
  
#信息收集-小程序抓包-Proxifier&BurpSuite联动  
1、抓包工具的证书要安装到系统的受信任和中间两个地方  
2、Proxifier规则设置里面一定要确定默认规则没有走代理  
-对抓到的IP或域名进行Web安全测试  
-对抓到的IP或域名进行API安全测试  
-对抓到的IP或域名进行端口服务测试  
  
#信息收集-小程序逆向-解包反编译&动态调试&架构  
对源码架构进行分析  
-更多的资产信息  
-敏感的配置信息  
-未授权访问测试  
-源码中的安全问题  
https://github.com/r3x5ur/unveilr  
https://github.com/Ackites/KillWxapkg  
https://github.com/biggerstar/wedecode  
https://github.com/eeeeeeeeee-code/e0e1-wx  
https://developers.weixin.qq.com/miniprogram/dev/devtools/stable.html

涉及资源：[资源下载地址](https://docs.qq.com/doc/DQ3Z6RkNpaUtMcEFr)