静态代码扫描在360项目的实践

Qtest - 刘俊

360
www.360.cn

Test
服 务 端 测 试

目录

- 需求？
- 解决方案
- 效果展示

需求？

一行代码的故事

```
public static void b(Context paramContext)
{
  if (paramContext == null)
    return;
  try
  {
    ActivityManager localActivityManager = (ActivityManager)paramContext.getSystemService("activity");
    localActivityManager.killBackgroundProcesses("me.ele.napos");
    localActivityManager.killBackgroundProcesses("com.dianping.ta");
    localActivityManager.killBackgroundProcesses("com.taobao.tongcheng");
    localActivityManager.killBackgroundProcesses("com.baidu.lbs.commercialism");
    return;
  }
  catch (Exception localException)
  {
    localException.printStackTrace();
```

需求？

放心吧
老板！

老板：不能出现

需求？

- 项目紧张，增加工作量
- 牵一发动全身，改出问题谁负责
- 别人写的代码，我不清楚
- 历史问题
- 这个事儿优先级不高，以后...

需求？

不同的声音

痛点	需求
• 假设性问题太多 • 规则无法定制 • 使用麻烦 • 报告可读性差 • 只说问题，没有方案	• 精准 • 快速 • 分级 • 共享 • 历史查询 • 易用 • 白名单

解决方案

名称	是否开源	功能列表	集成方式	支持语言	优点
Lint	SDK内置工具	无效布局，多重嵌套等 未使用的冗余资源文件 数组资源大小不一致 图标缺失、重复、错误尺寸 AndroidManifest.xml中的错误	命令行 IDE CI	java	专注于安卓的特性检测
Checkstyle	开源： http://checkstyle.sourceforge.net/	检查类及方法的 Javadoc 注释 检查命名是否符合命名规范 检查文件是否以某些行开头 检查 Import 语句是否符合定义规范 即检查类、方法等代码块的行数 检查空白符，如 tab，回车符等 修饰符号的检查，如修饰符的定义顺序 检查是否有空块或无效块 检查重复代码，条件判断等问题	命令行 IDE CI	java	基于sun公司和google推出的java编程规范来检测，严谨和完善
FindBugs	开源： http://findbugs.sourceforge.net/	Bad practice：常见代码错误 可能导致错误的代码：如空指针引用等 国际化相关问题：如错误的字符串转换 可能受到的恶意攻击：如访问权限修饰符的定义等 多线程的正确性：如多线程编程时常见的同步，线程调度问题。 运行时性能问题：如由变量定义，方法调用导致的代码低效问题。	命令行 IDE GUI CI	java	只关注潜在缺陷和性能问题，基于LGPL协议，无需运行就能对代码解析，从bytecode入手进行检查,对空指针检测有独到之处
PMD	开源： https://pmd.github.io	检查潜在代码错误，如空 try/catch/finally/switch 语句 检查未使用的变量，参数，方法 检查不必要的 if 语句，可被 while 替代的 for 循环 检查重复的代码 检查在循环体内实例化新对象 检查 Connect，Result，Statement 等资源使用之后是否被关闭掉	命令行 IDE GUI CI	Java, C, C++, C#, PHP, Ruby, Fortran, JavaScript, PLSQL, Apache Velocity, Ruby, Scala, Objective C, Matlab, Python, Go	语言扩展性出色，资源关闭和回收，循环创建对象等问题也是开发人员比较关注的问题

解决方案

工具名称：  **火 线（FireLine）**

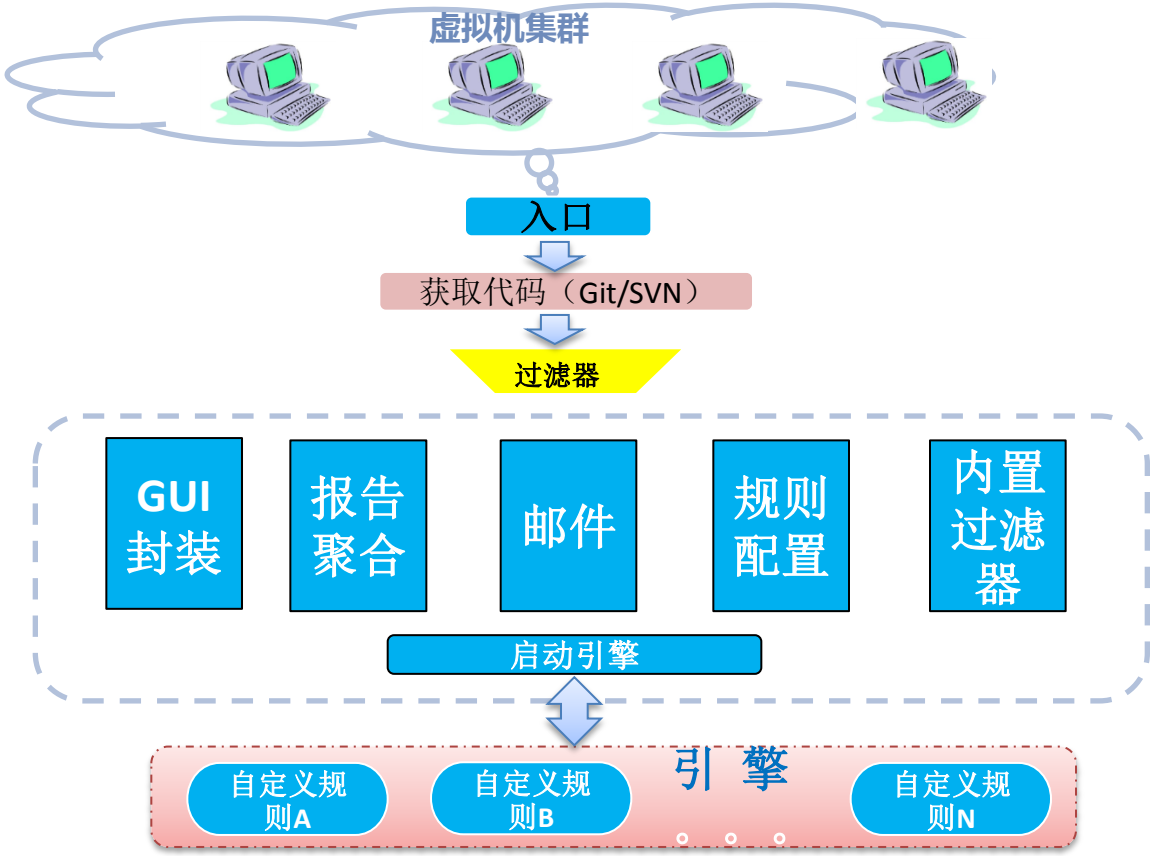
核心规则：奇虎Android开发安全SDL（信息安全部门支持）

应用场景：静态代码扫描 (缺陷模式匹配+字节码解析)

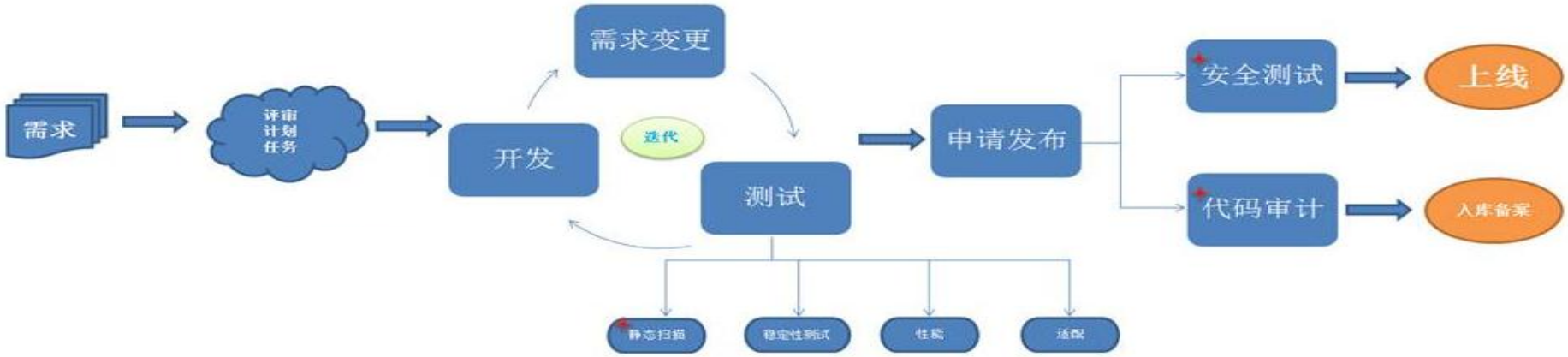
运行方式：GUI、Command、Service

官网地址：magic.360.cn

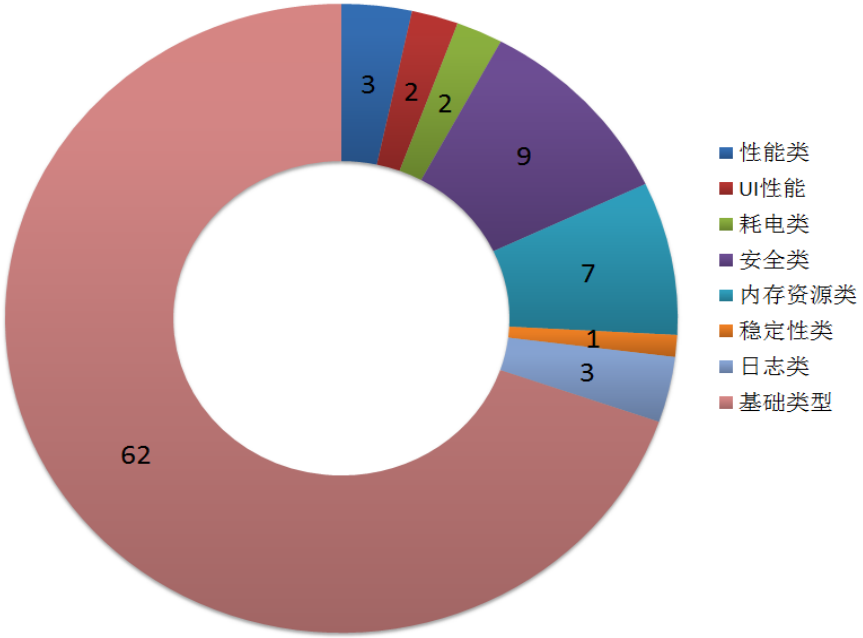
火线的结构


虚拟机集群
入口
获取代码（Git/SVN）
过滤器
GUI封装
报告聚合
邮件
规则配置
内置过滤器
启动引擎
自定义规则A
自定义规则B
引擎
自定义规则N

火线得到公司流程的保障



火线的检查类型



解决方案

类型	常见问题
基础规则	1. 错位的空判断：if (!string.equals("") && string!=null) 2. 破坏空判断：if (string!=null || !string.equals("")) 3. 避免使用 IP硬编码 4. 应该在方法开始处调用super方法 5. 硬编码SDCard路径 6. 资源对象没有在finally中关闭 …
日志类规则	1. 日志敏感信息输出 2. 日志函数中进行变量赋值或调用方法 3. 日志输出没有包裹在DEBUG开关中
安全类规则	1. WebView未禁止访问本地文件系统 2. 敏感信息安全 3. Android组件安全 4. SQL注入 5. 目录遍历漏洞 …
内存管理类	1. IO流关闭 2. Cursor关闭 3. Bitmap释放 4. 资源对象关闭：Connection,Statement,和ResultSet对象 …
性能优化类	
其他类型	
稳定性规则	
UI性能规则	
耗电类规则	

解决方案

截止到5月11日火线的数据：

数据指标	本周数据
工具累计运行天数	150天
累计执行任务次数	6597次
已覆盖的项目数	329个任务
已扫描代码总行数	约1.7亿行
已扫描总文件数	约47万个
单个项目平均扫描时间	31.58秒
每千行代码扫描时间	0.459秒
已使用工具的开发人员数量	137位用户

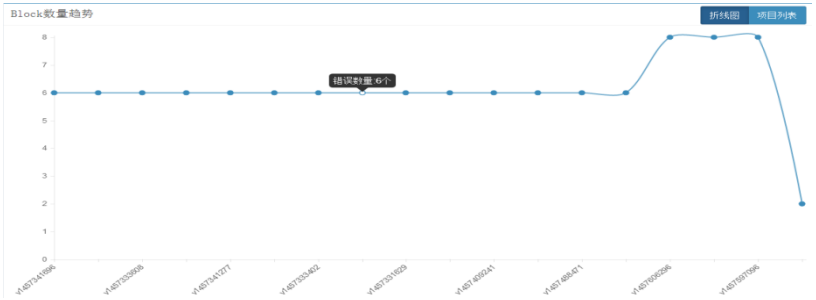
解决方案

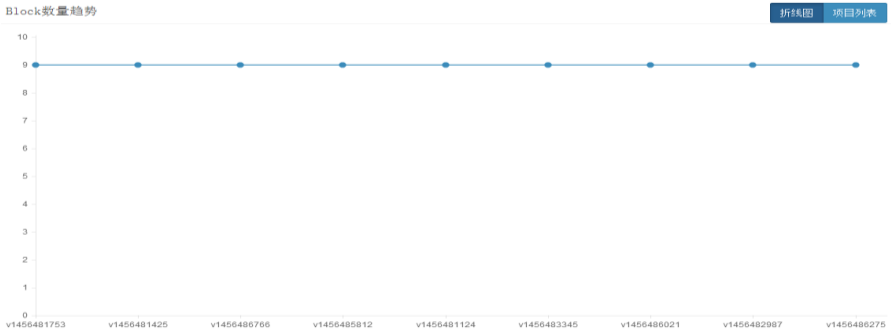
目前扫描项目：

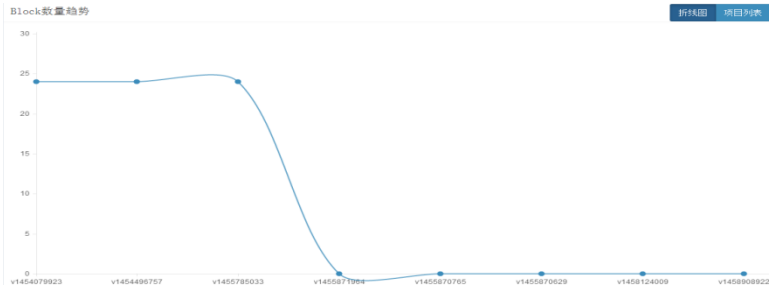
- 360手机卫士
- 360手机助手
- 360手机浏览器
- 360游戏大厅
- 360云盘
- 360桌面
- 360 Security
- 360智能摄像机
- 360 OS
- 360免费WiFi
- 路由器卫士
- 360行车记录仪
- 360商城
- 手心输入法
- 360搜索

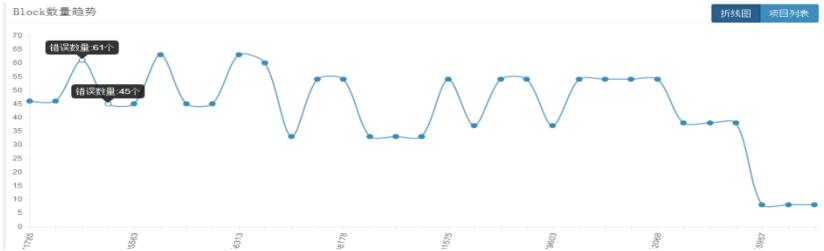
解决方案

数据中心的特征：



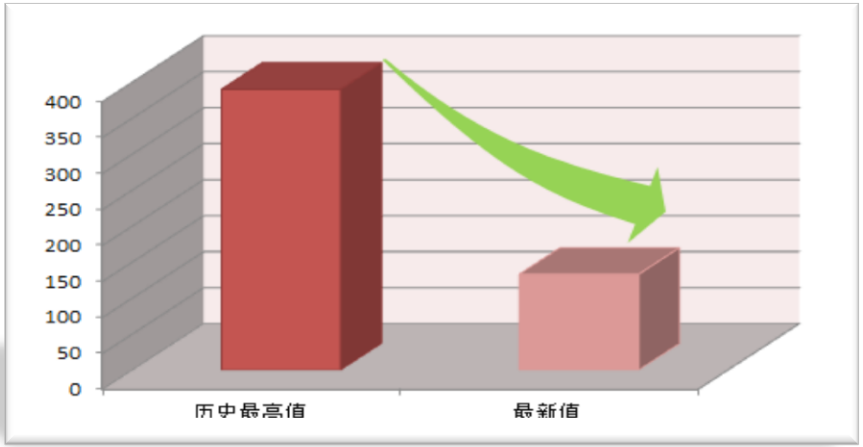






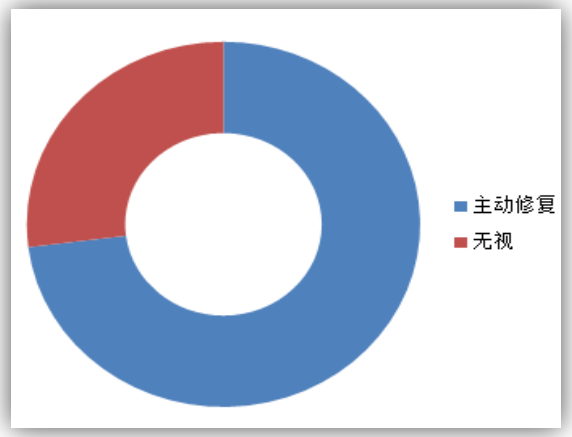
解决方案

去重后数据显示，项目总的block数量从最高时候的390降低到133，减少了**66%**，整体代码质量呈**上升趋势!**



73%的项目针对 Block 错误有修复行为

27%的项目对 Block 错误选择无视



效果展示

代码展示（部分代码改编自公司无线产品项目源码）

扫描结果准确性：

问题	FindBugs	Lint	PMD	火线	缺陷总计
SQL注入检测				√	FindBugs：**36**个 Lint：**5**个 PMD：**331**个 火线：**33**个
IP硬编码			√	√	
空指针	√			√	
变量定义未使用			√	√	
多余的if语句			√	√	
对象比较用==	√		√	√	
Stream流没有正常关闭				√	
Stream流没有在finally块中关闭	√			√	
日志输出了敏感信息				√	
日志输出没有开关项控制				√	
I/O对象未关闭				√	
空的 try/catch/finally 块			√	√	
Sdcard硬编码	√	√		√	
Cursor对象关闭				√	
目录遍历漏洞检测				√	
问题命中率	26.66%	6.66%	33.33%	100%	

效果展示

测试报告的可读性

FindBugs报告

FindBugs Report

Project Information

Project:

FindBugs version: 3.0.1

Code analyzed:

- D:\workspace\AndroidDemo\bin\classes

Metrics

2819 lines of code analyzed, in 36 classes, in 3 packages.

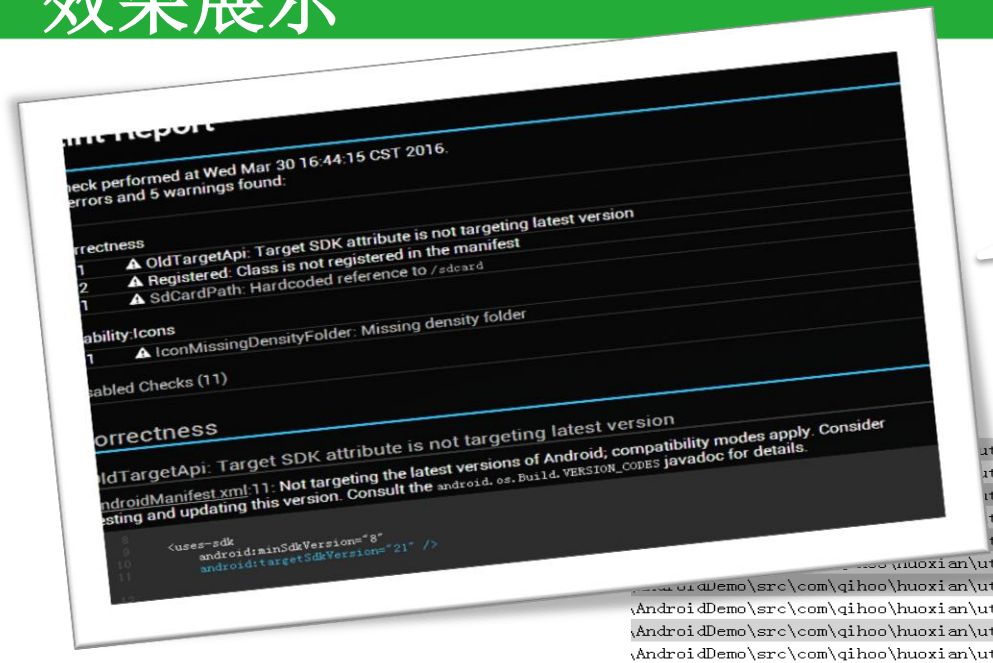
Metric	Total	Density*
High Priority Warnings	4	1.42
Medium Priority Warnings	32	11.35
Total Warnings	**36**	**12.77**

(Defects per Thousand lines of non-commenting source statements)*

Correctness Warnings

Code	Warning
NP	Null pointer dereference of temp in com.example.androiddemo.MainActivity.demo(String[], String[], String) Bug type NP_ALWAYS_NULL (click for details) In class com.example.androiddemo.MainActivity In method com.example.androiddemo.MainActivity.demo(String[], String[], String) Value loaded from temp Dereferenced at MainActivity.java:[line 73]
NP	Possible null pointer dereference of FuncSp.mDB in com.qihoo.huoxian.utils.FuncSp.getCursor(Context, String)
UwF	Unwritten field: com.qihoo.huoxian.utils.FuncSp.mDB Bug type UWF_UNWRITTEN_FIELD (click for details) In class com.qihoo.huoxian.utils.FuncSp Field com.qihoo.huoxian.utils.FuncSp.mDB At FuncSp.java:[line 91]
UwF	Unwritten field: com.qihoo.huoxian.utils.MyProvider.values Bug type UWF_UNWRITTEN_FIELD (click for details) In class com.qihoo.huoxian.utils.MyProvider Field com.qihoo.huoxian.utils.MyProvider.values At MyProvider.java:[line 100]

效果展示



Lint报告

PMD报告

utils\FuncSp.java	86	A method should have only one exit point, and that
utils\FuncSp.java	90	Local variable 'sql' could be declared final
utils\FuncSp.java	91	Avoid empty if statements
tils\FuncSp.java	94	
tils\FuncSp.java	94	Found 'DD'-anomaly for variable 'c' (lines '94'-'9
huoxian\utils\FuncSp.java	95	A catch statement should never catch throwable sin
dDemo\src\com\qihoo\huoxian\utils\FuncSp.java	96	Assigning an Object to null is a code smell. Consi
AndroidDemo\src\com\qihoo\huoxian\utils\FuncSp.java	101	Avoid variables with short names like c
AndroidDemo\src\com\qihoo\huoxian\utils\FuncSp.java	101	Parameter 'c' is not assigned and could be declare
AndroidDemo\src\com\qihoo\huoxian\utils\FuncSp.java	102	Comment is too large: Line too long
AndroidDemo\src\com\qihoo\huoxian\utils\FuncSp.java	107	Avoid catching generic exceptions such as NullPoin
AndroidDemo\src\com\qihoo\huoxian\utils\FuncSp.java	107	Avoid empty catch blocks
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	10	All methods are static. Consider using a utility c abstract to silence this warning.
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	11	Avoid variables with short names like c
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	11	To avoid mistakes add a comment at the beginning o
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	11	Use explicit scoping instead of the default packag
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	16	Avoid catching generic exceptions such as NullPoin
AndroidDemo\src\com\qihoo\huoxian\utils\IOUtil.java	16	Avoid empty catch blocks
AndroidDemo\src\com\qihoo\huoxian\utils\MyProvider.java	6	Avoid unused imports such as 'android.os.Bundle'
AndroidDemo\src\com\qihoo\huoxian\utils\MyProvider.java	7	Avoid unused imports such as 'android.app.Activity
AndroidDemo\src\com\qihoo\huoxian\utils\MyProvider.java	18	Avoid unused imports such as 'android.view.Menu'
AndroidDemo\src\com\qihoo\huoxian\utils\MyProvider.java	24	Each class should declare at least one constructor
AndroidDemo\src\com\qihoo\huoxian\utils\MyProvider.java	24	This class has too many methods, consider refactor

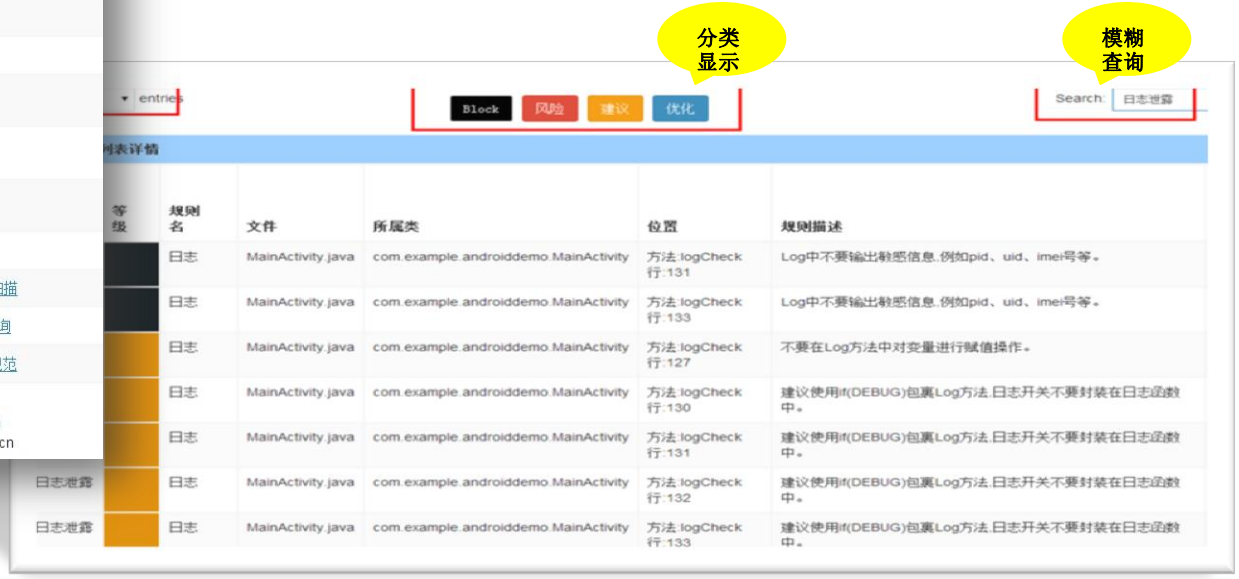
效果展示

火线汇总报告	
扫描项目路径	D:\asWrokSpace\FirstDemo
开始时间	2016-05-11 18:30:14
结束时间	2016-05-11 18:30:22
测试耗时	8秒258毫秒
项目名称	test
提交人	
扫描文件数	32个
扫描代码行数	15612行
Block错误数(必须修改*)	1
安全风险数(需安全部门确认)	8
其他错误数(建议修改)	3 3
火线官网地址	火线官网
在线扫描地址	开发可上传APK进行在线扫描
数据中心地址	提供项目历次扫描结果查询
参考文献地址	奇虎安卓安全 - SDL安全规范
问题咨询联系人	火线开发人员： 刘俊：liujun-iri@360.cn 袁伟：yuanwei3-iri@360.cn

火线报告

分类显示

模糊查询



效果展示

合适你的才是最好的

谢 谢！

安全第一