思客云

让“开发者爱上安全测试”

软件安全咨询师:王 宏
hwang@secureyun.cn
http://www.secureyun.cn

从 **安全开发** 到 **安全测试**

SecDevOps

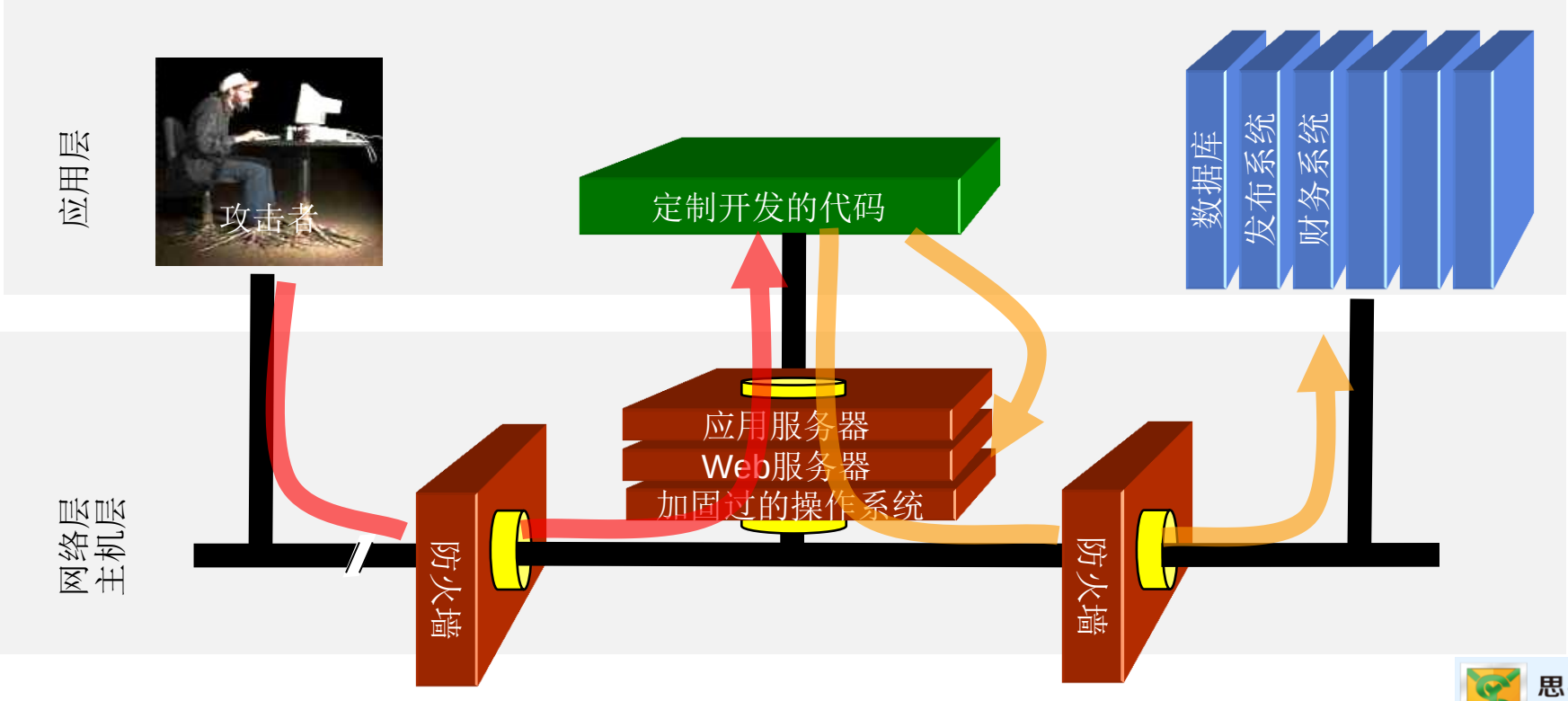
为什么需要源代码安全测试？

----Software is incomplete until it’s secure

（软件：唯有安全，方为完整）

应用层直接暴露在威胁之下

防火墙、入侵检测、主机安全加固可以有效防御针对主机和网络的攻击，
但是对应用层却几乎束手无策



2016年规模较大的信息泄露事件

2015年：

12月末	美国1.9亿选民信息泄露

2016年：

1月	俄罗斯邮件网站Mail.ru约5700万登录凭证在网上出售
4月	5000万土耳其公民信息泄露
4月	5500千万菲律宾选民信息泄露
4月	9340万墨西哥选民个人信息数据库曝光
5月	1.17亿LinkedIn账户登录信息泄露
5月	4000万成人社交网站Fling用户的凭证在暗网售卖
6月	俄罗斯社交网站VK.com1亿登录凭证被盗
8月	俄罗斯搜索引擎Rambler约1亿用户信息网上曝光
9月	雅虎5亿账户信息泄露
10月	MongoDB 5800万商业用户信息泄露
12月	影片分享网站Dailymotion 8520万用户名及邮件泄露
12月	雅虎确认一起早在2013年的账户信息泄露，这次的数字是10亿

2016年漏洞类型统计

漏洞类型	漏洞数量	占比
缓冲区溢出	1207	15.31%
权限许可和访问控制	853	10.82%
信息泄露	842	10.68%
跨站脚本	573	7.27%
输入验证	552	7.00%
资源管理错误	171	2.17%
SQL注入	135	1.71%
数字错误	119	1.51%
跨站请求伪造	118	1.50%
路径遍历	89	1.13%

代码安全漏洞谁的错？

软件开发人员是软件的缔造者，也是软件安全漏洞的制造者，应该为软件安全漏洞负责。

——王宏总结

SQL 注入的代码案例

```
import java.sql.*;

public class SQLInjection {
public static void main (String args[]) {
   Connection conn = null;
   try {
        String userName = args [0];
        String passwd   = args [1];
        String query = "select uname, passwd from users where uname = '"+userName+"'";
        conn = DriverManager.getConnection ("jdbc:odbc:logistics", "admin", "letmein");
        Statement stmnt = conn.createStatement ();
        ResultSet rs = stmnt.executeQuery (query);
        while ( rs.next() ) {
            ...
            }
        rs.close ();
        stmnt.close ();
        conn.close ();
       }
   catch (SQLException err) {
     err.printStackTrace ();
     }
   }
}
```

不正确地使用不可信数据
拼接（+）

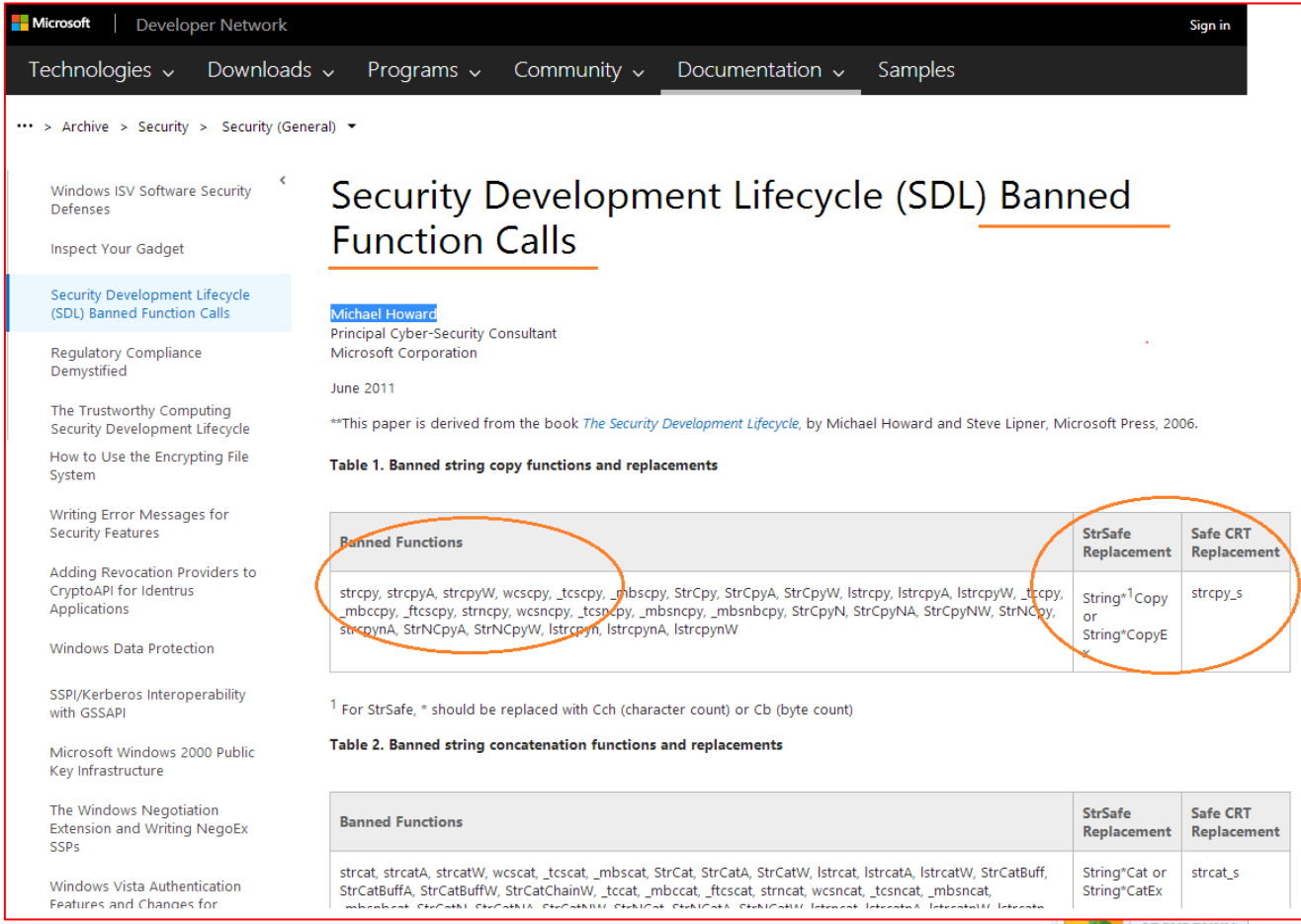
Q: 是谁教我们(开发者)这么写代码的？

Bad API

- Microsoft 公司出具 Banned Function Call

——*Michael Howard*

- https://msdn.microsoft.com/en-us/library/bb288454.aspx



Microsoft | Developer Network — Sign in

Technologies | Downloads | Programs | Community | Documentation | Samples

... > Archive > Security > Security (General)

Windows ISV Software Security Defenses
Inspect Your Gadget
Security Development Lifecycle (SDL) Banned Function Calls
Regulatory Compliance Demystified
The Trustworthy Computing Security Development Lifecycle
How to Use the Encrypting File System
Writing Error Messages for Security Features
Adding Revocation Providers to CryptoAPI for Identrus Applications
Windows Data Protection
SSPI/Kerberos Interoperability with GSSAPI
Microsoft Windows 2000 Public Key Infrastructure
The Windows Negotiation Extension and Writing NegoEx SSPs
Windows Vista Authentication Features and Changes for

Security Development Lifecycle (SDL) Banned Function Calls

Michael Howard
Principal Cyber-Security Consultant
Microsoft Corporation

June 2011

**This paper is derived from the book *The Security Development Lifecycle*, by Michael Howard and Steve Lipner, Microsoft Press, 2006.

Table 1. Banned string copy functions and replacements

Banned Functions	StrSafe Replacement	Safe CRT Replacement
strcpy, strcpyA, strcpyW, wcscpy, _tcscpy, _mbscpy, StrCpy, StrCpyA, StrCpyW, lstrcpy, lstrcpyA, lstrcpyW, _tccpy, _mbccpy, _ftcscpy, strncpy, wcsncpy, _tcsncpy, _mbsncpy, _mbsnbcpy, StrCpyN, StrCpyNA, StrCpyNW, StrNCpy, strcpynA, StrNCpyA, StrNCpyW, lstrcpyn, lstrcpynA, lstrcpynW	String*[1]Copy or String*CopyEx	strcpy_s

[1] For StrSafe, * should be replaced with Cch (character count) or Cb (byte count)

Table 2. Banned string concatenation functions and replacements

Banned Functions	StrSafe Replacement	Safe CRT Replacement
strcat, strcatA, strcatW, wcscat, _tcscat, _mbscat, StrCat, StrCatA, StrCatW, lstrcat, lstrcatA, lstrcatW, StrCatBuff, StrCatBuffA, StrCatBuffW, StrCatChainW, _tccat, _mbccat, _ftcscat, strncat, wcsncat, _tcsncat, _mbsncat,	String*Cat or String*CatEx	strcat_s

天使？恶魔？

- 让“大脑”的中“小人儿” 斗争起来吧！

英特尔尝试在CPU级别干掉缓冲区溢出

2016-06-20 09:56　来源：安全牛网　字号： 大 中 小

英特尔正在推广一项很灵活的技术。该技术可以在处理器层面上阻挡恶意软件感染，其相关细节已经在上周四发表。实际上是这样：英特尔称之为控制流强制技术(Control-flow Enforcement Technology, CET)尝试阻挠会使用面向返回编程(Return-orientated programming, ROP)和面向跳转编程(Jump-orientated programming, JOP)的漏洞利用代码。

总结

犯错（产生漏洞）是不可避免的！

但：错误是可以规避的！

知行合一

Q:执行代码安全测试的最大阻力是什么？

- 没有专业的安全人员
- 不懂代码安全漏洞知识
- 没有安全测试工具或测试工具不好用
- 领导不懂安全测试或不重视安全测试

- **开发者不配合，安全漏洞不认**

为什么开发者不配合安全测试？

因为……

- **My Code**

```
if( isset($db_key_config[$appName]) )
{
    if( is_array($db_key_config[$appName]) )
    {
        if( !empty($keyName) )
        {
            if( in_array($keyName,$db_key_config[$appName]) )
            {
                if( strlen($keyValue) < 10240 )
                {
                    $ttsObj = @memcache_connect($TTServer_config['
                    if( $ttsObj )
                    {
                        $dbStr = $ttsObj->get($appKey);
                        if( !empty($dbStr) )
                        {
                            $dbArr = json_decode($dbStr,true);
                            if( !empty($keyValue) )
                            {
                                if( $backCode == -1 )
                                {
                                    $dbArr[$keyName] = $keyValue;
```

=

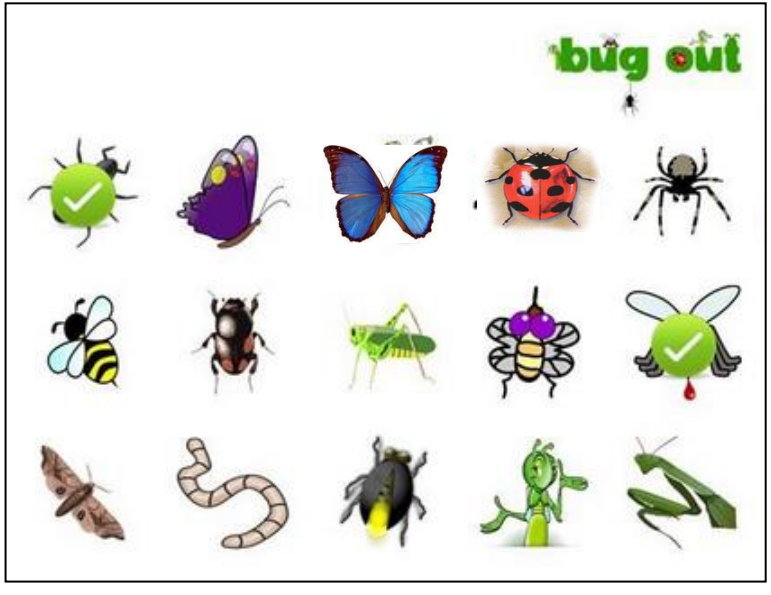
- **My Baby**

你说他有毛病？

上线前的关键时刻：讨人厌的安全城管

我要的是真正的BUG！

- 由于测试人员、测试工具、测试环境等种种原因，测试结果中有较大的误报，大量的无效信息



于是……

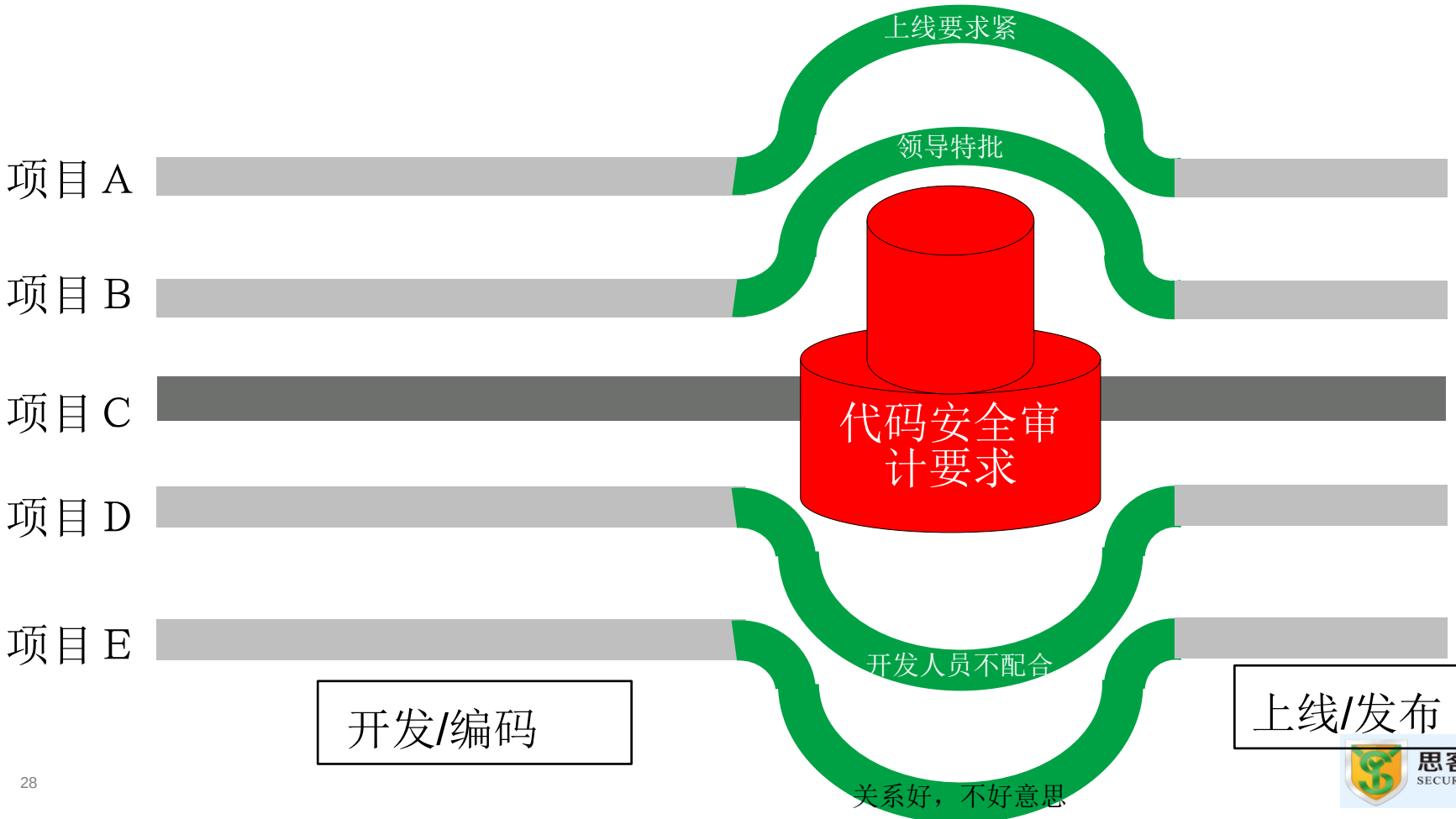
代码安全测试到底该由谁负责？

安全？开发？测试？

三个和尚没水喝？

制度＝＝一堆高大上的文档？

制度的不健全



审计方法问题：纯人工

- 简单的人+工具？

代码安全测试

环境搭建 审计工具学习 代码获取

代码版本 漏洞记录 安全知识

审计统计 漏洞审计

结果讨论

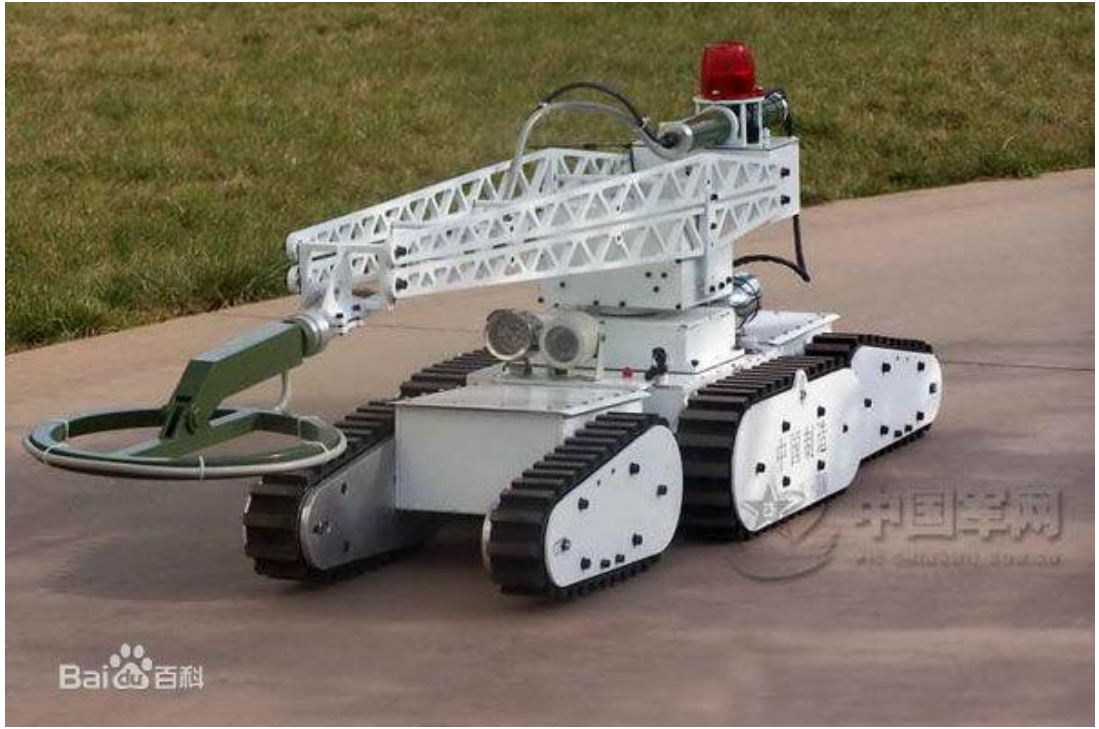
复测

代码测试员

最佳的代码安全测试是什么？

如何让“开发者”爱上代码安全测试？

至少需要一个全自动化的安全扫描平台



三个方面开展安全审计工作

人员

- 组建专业团队（权责分明）
- 专业的安全审计能力培训

制度

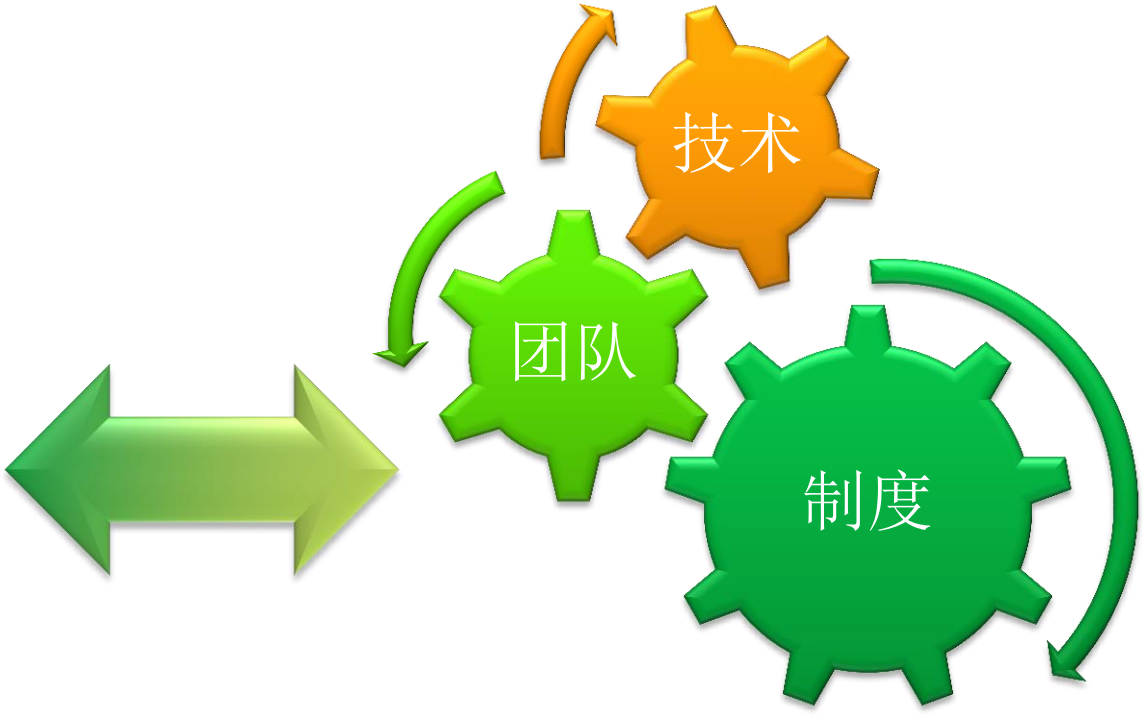
- 建立切实可行的安全审计规范
- 将制度工具化

技术

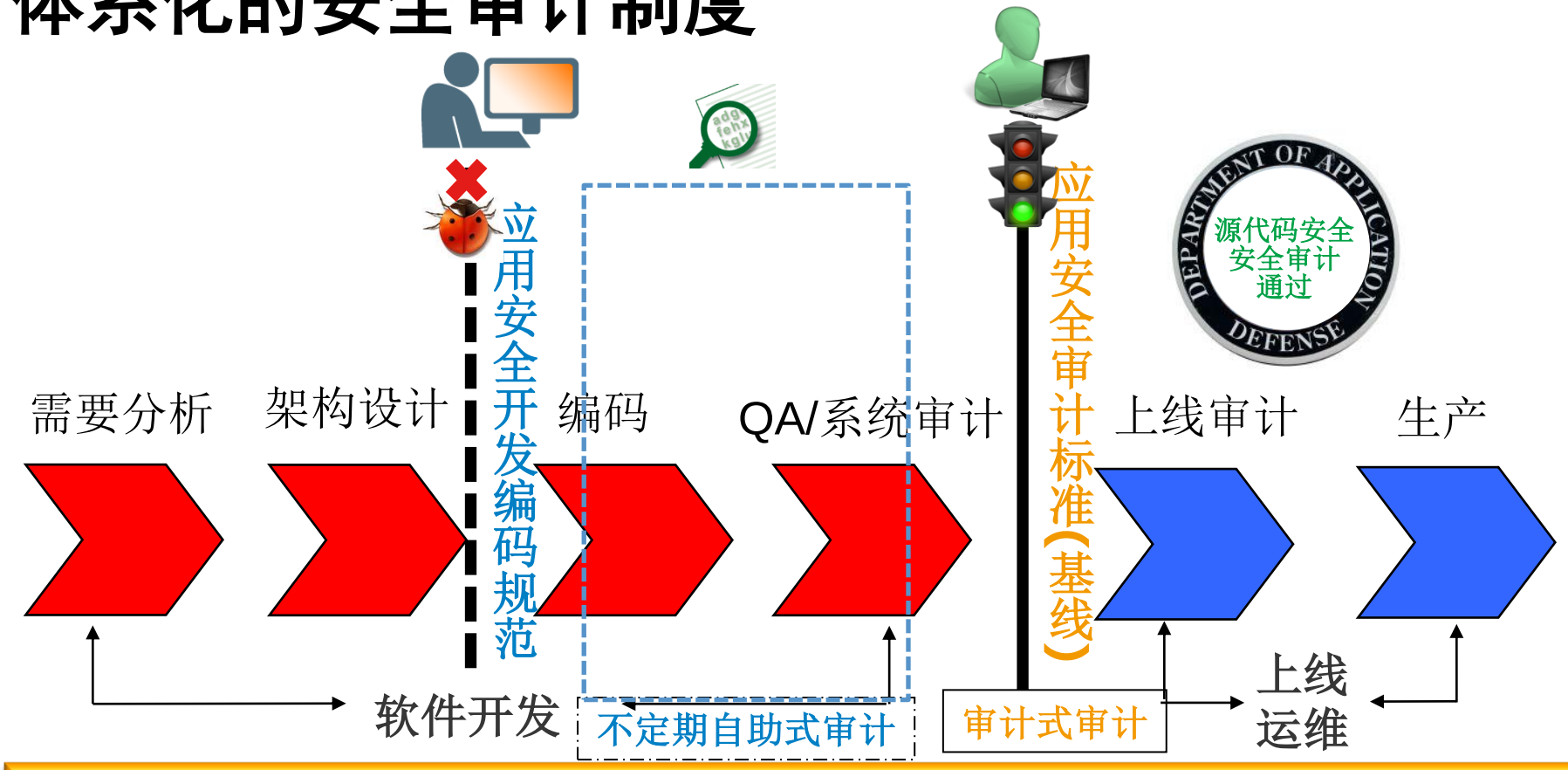
- 采用先进的安全审计技术和工具
- 让审计简单、方便
  人人爱

最佳的源代码安全审计方案是:

- **自动化** 审计工具
- **体系化** 审计制度
- **平台化** 审计技术
- **规范化** 审计管理
- **流程化** 审计团队



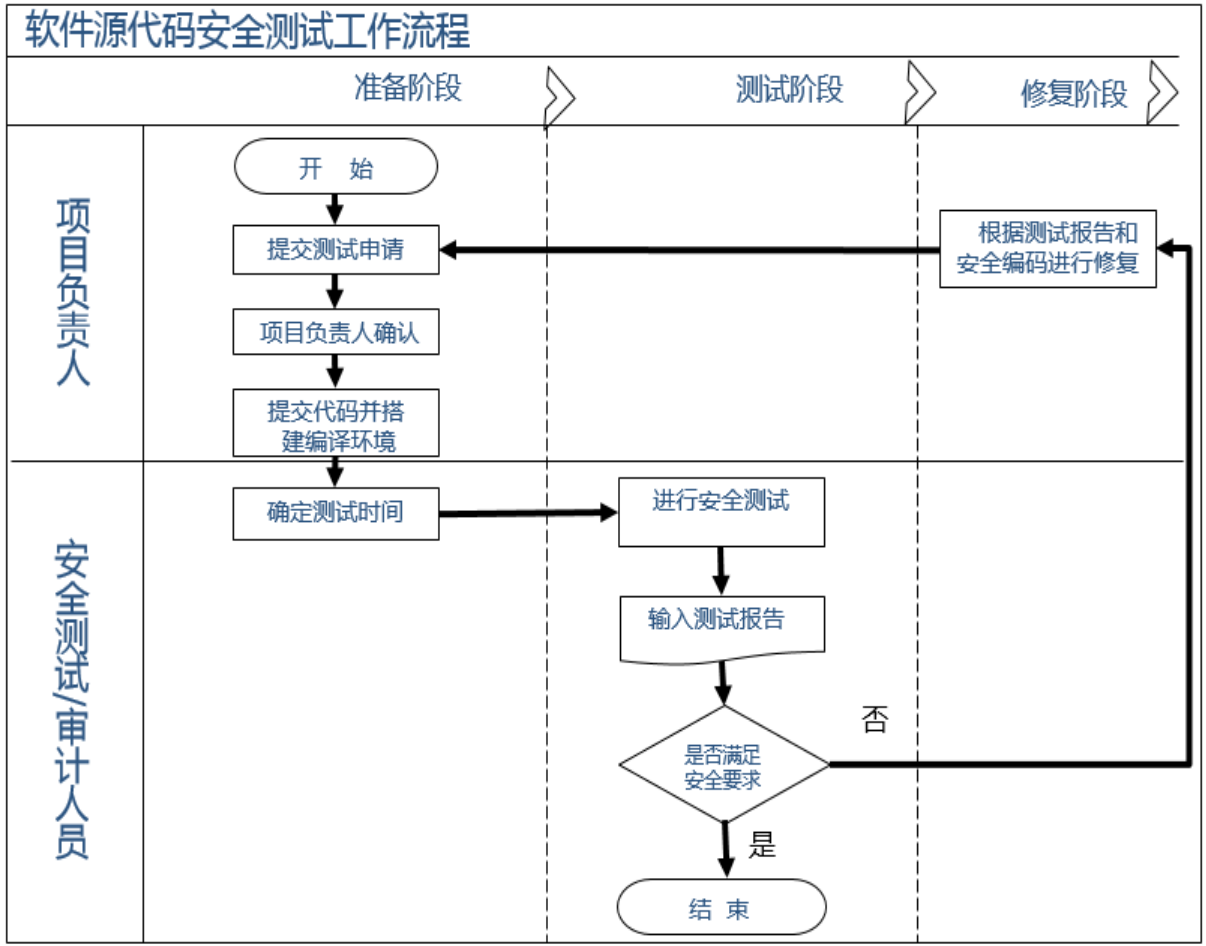
体系化的安全审计制度



软件源代码安全审计（找八哥）系统

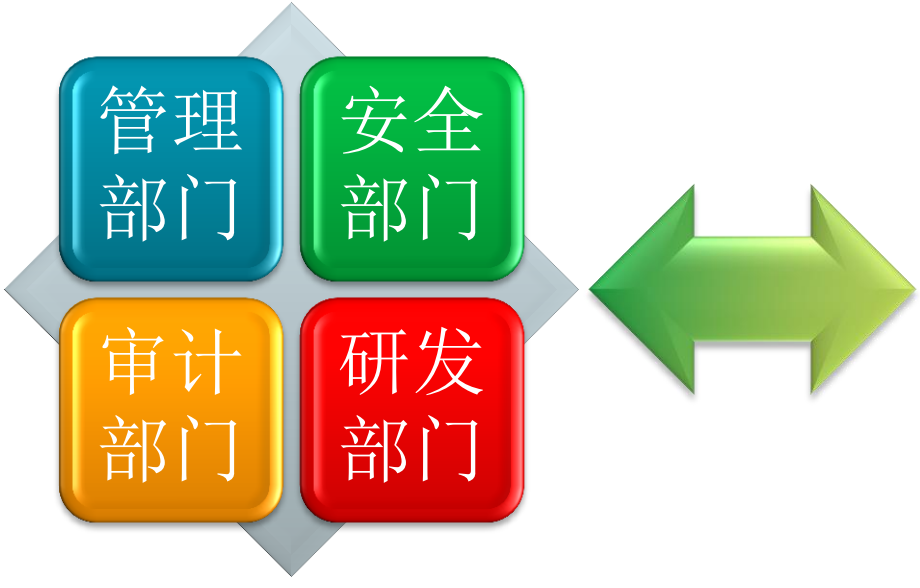
规范化的审计管理---完整的流程及文档

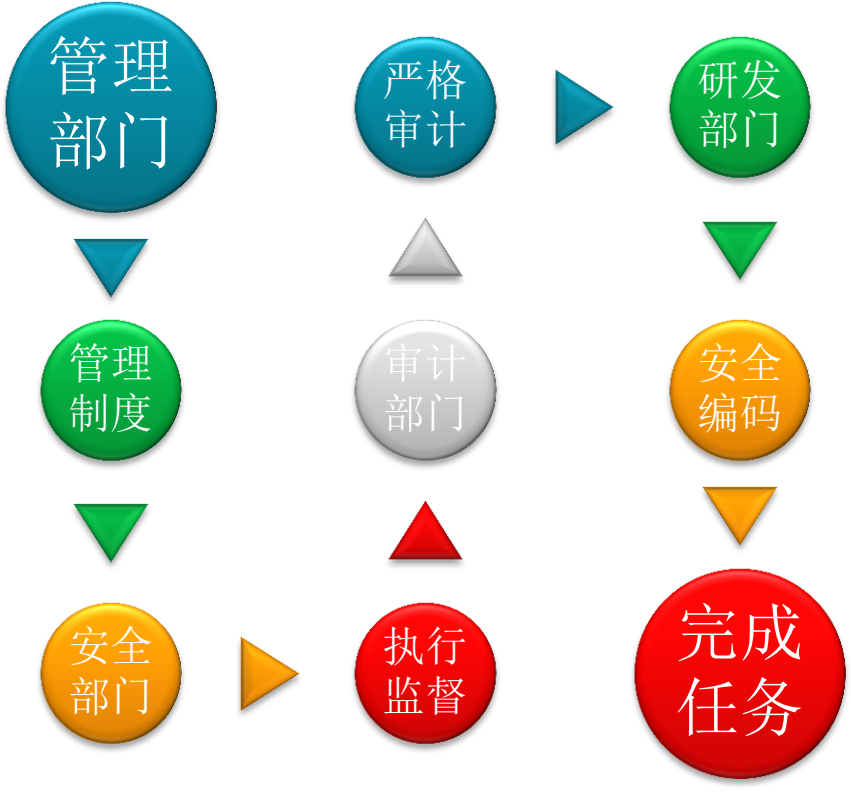
- 《软件源代码安全审计管理制度》
- 《软件源代码安全审计标准(基线)TOP10》
- 《软件源代码安全编码指南之TOP10》
- 《软件源代码安全审计指南之TOP10》
- 《软件源代码安全审计云平台操作手册》



平台化安全审计技术---找八哥系统

“流程化”的审计团队





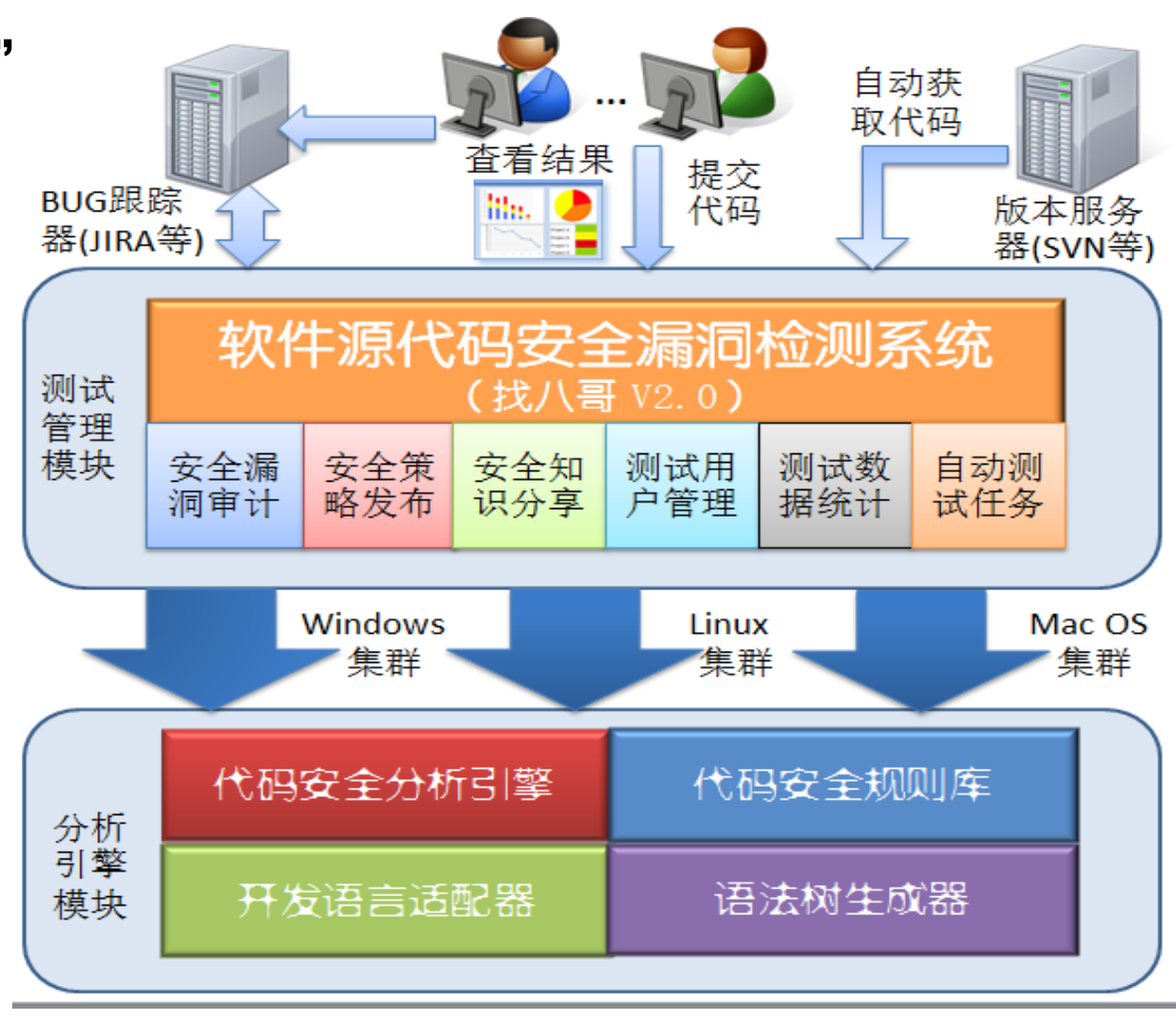
思客云“找八哥”能帮您带来什么？

--自动化 --体系化 --平台化 --规范化 --流程化

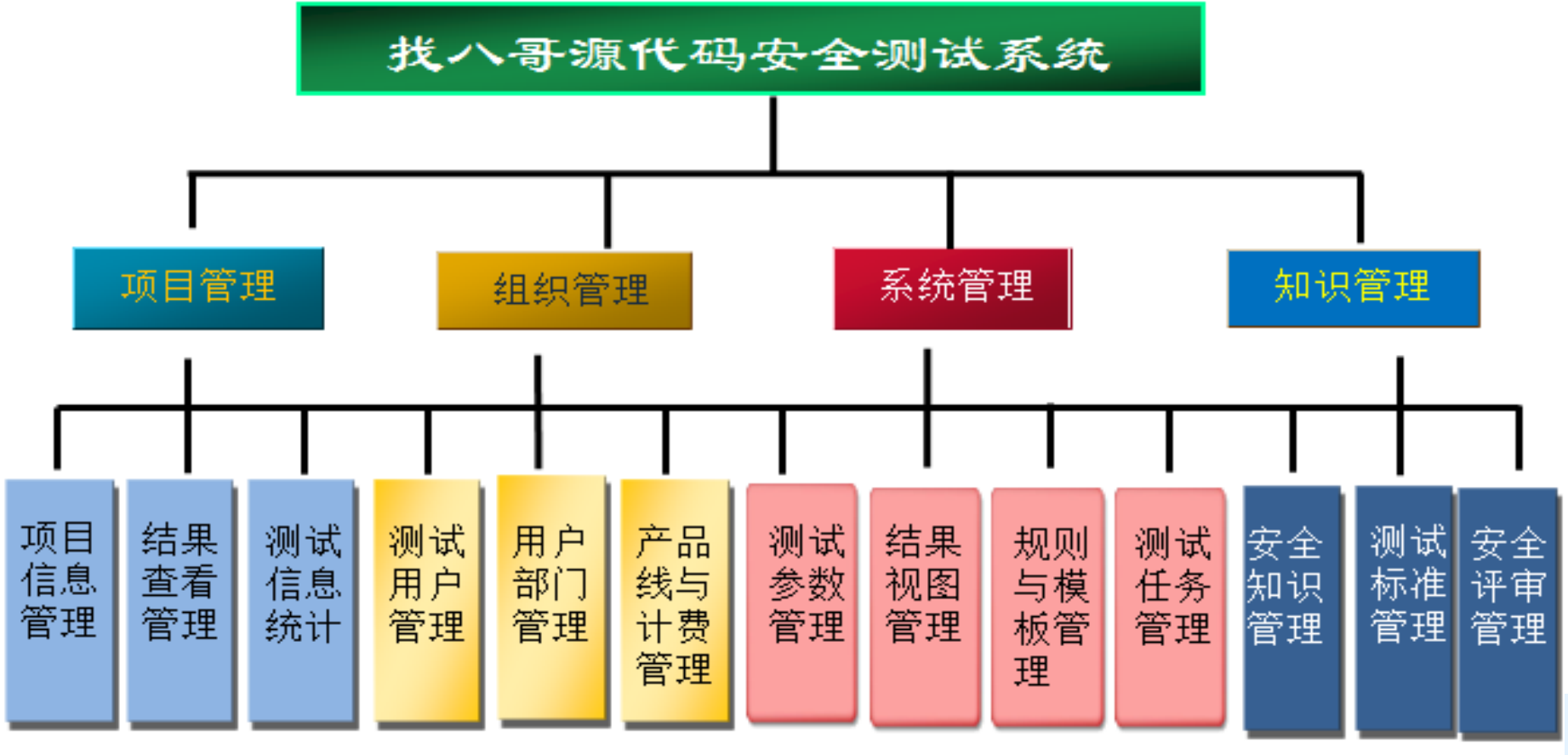
思客云---“找八哥”

ZHAOBUG

找八哥

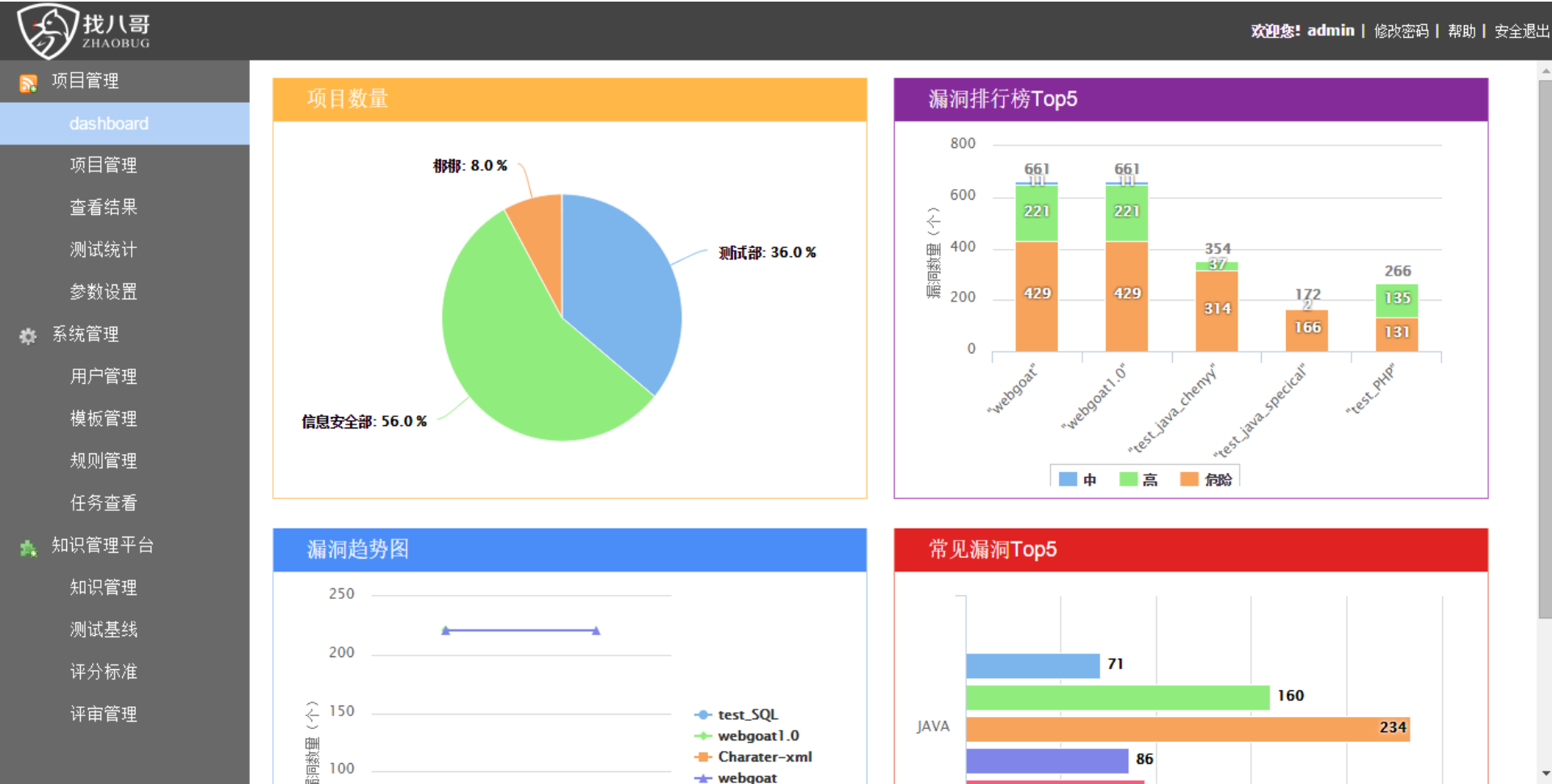


找八哥系统功能模块

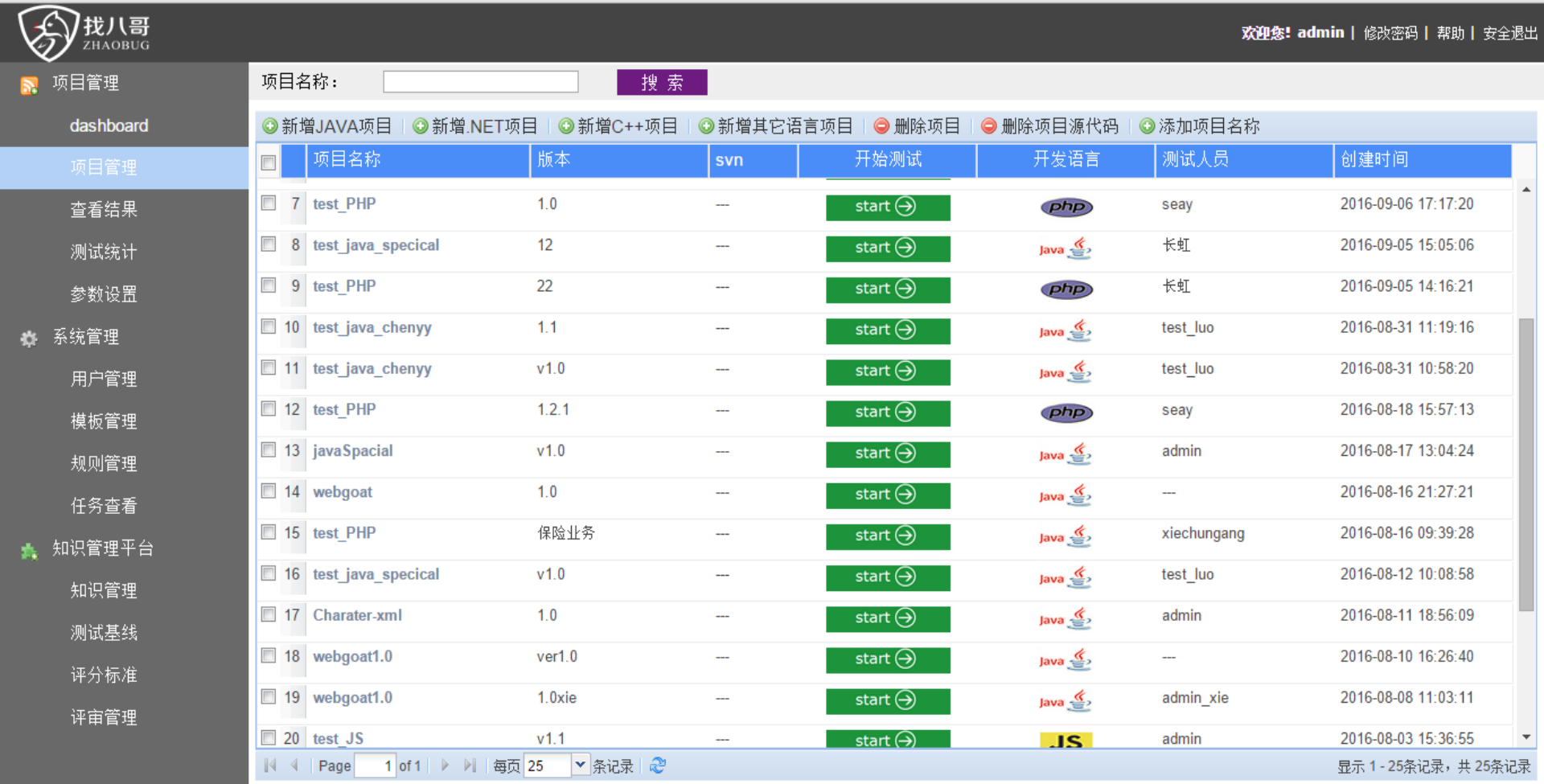

找八哥源代码安全测试系统
项目管理
组织管理
系统管理
知识管理
项目信息管理
结果查看管理
测试信息统计
测试用户管理
用户部门管理
产品线与计费管理
测试参数管理
结果视图管理
规则与模板管理
测试任务管理
安全知识管理
测试标准管理
安全评审管理

找八哥的优点

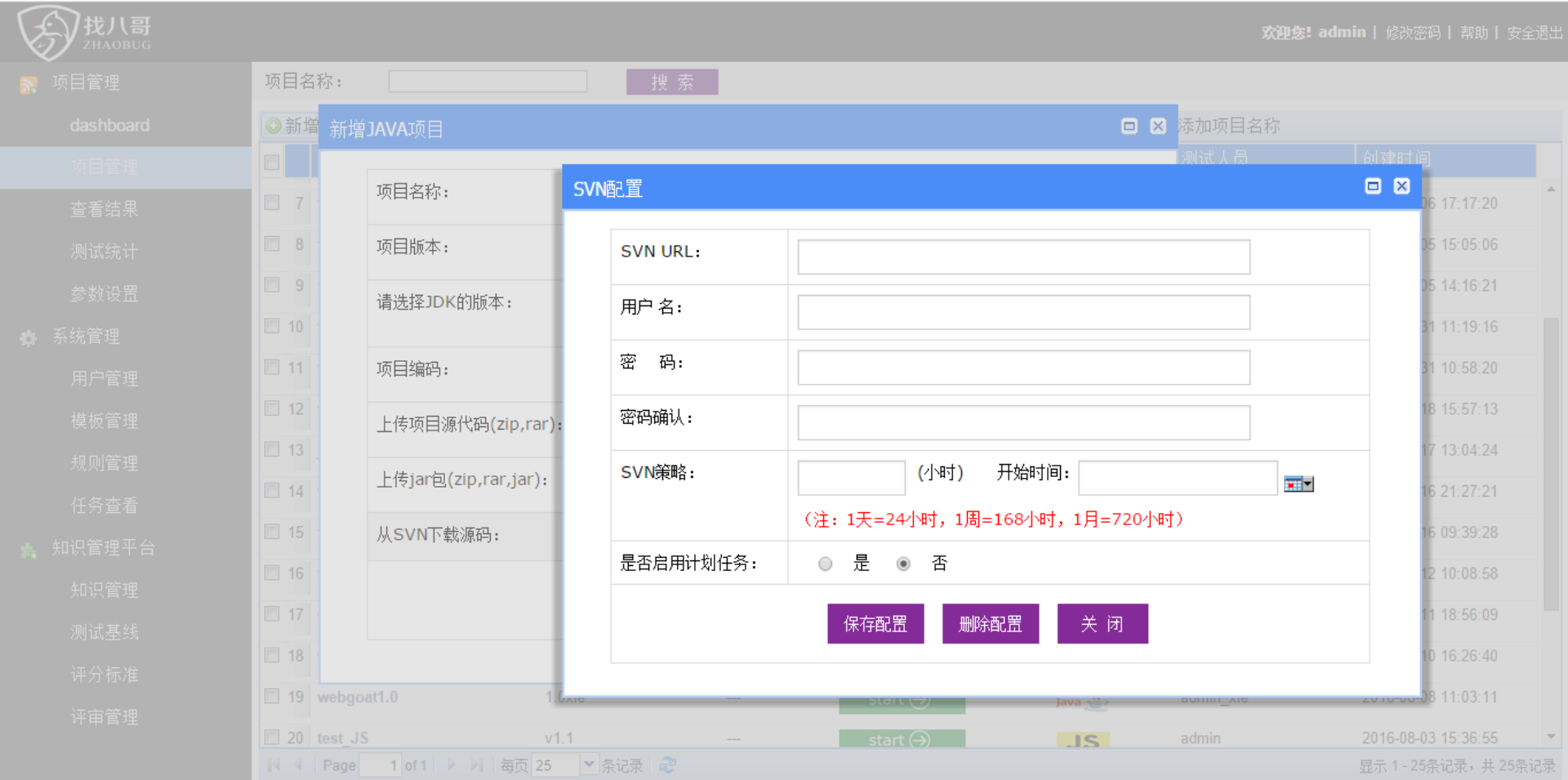
找八哥---Dashboard



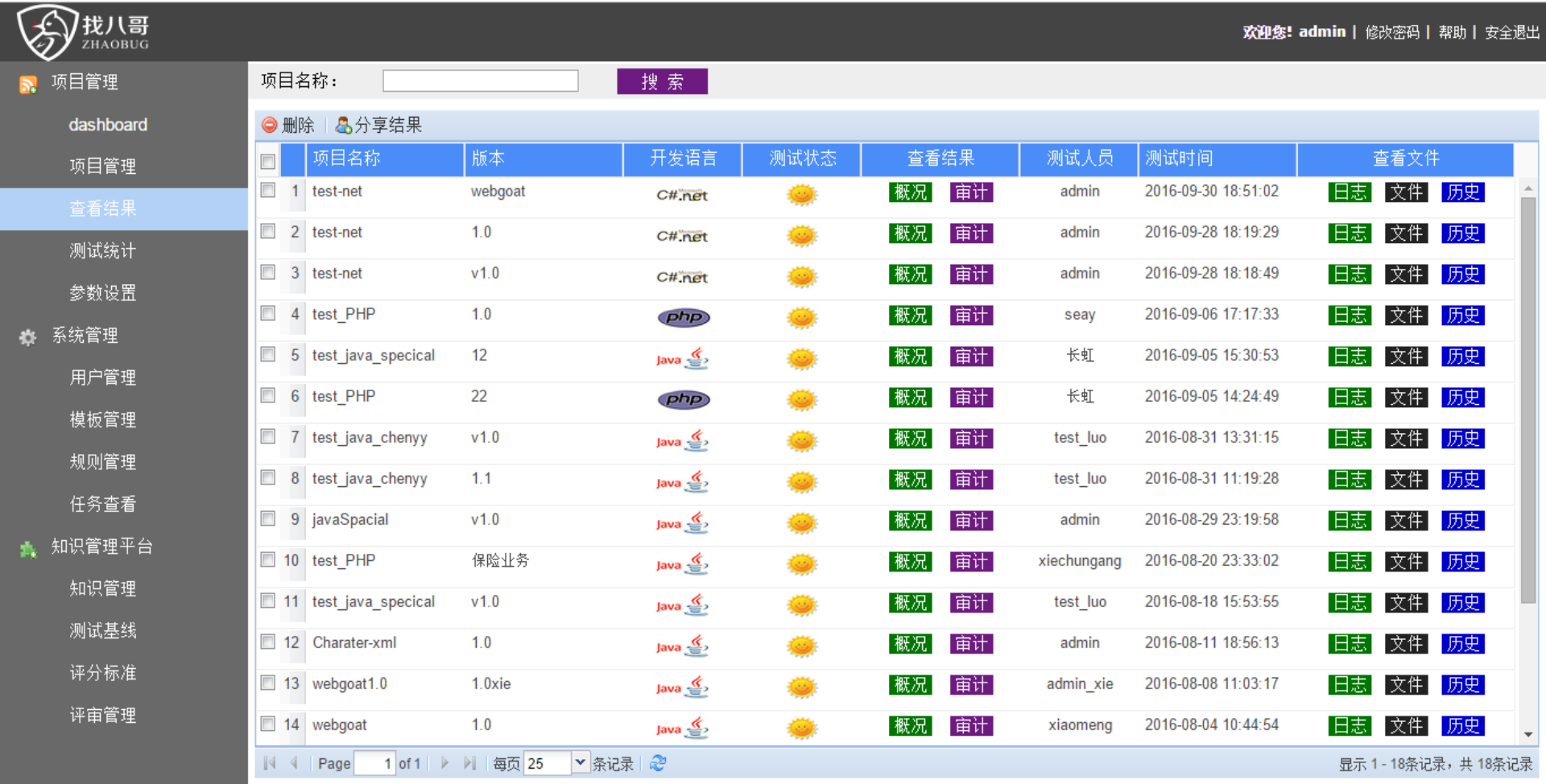
找八哥——项目管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
项目名称：
搜 索
新增JAVA项目 | 新增.NET项目 | 新增C++项目 | 新增其它语言项目 | 删除项目 | 删除项目源代码 | 添加项目名称
项目名称 | 版本 | svn | 开始测试 | 开发语言 | 测试人员 | 创建时间
7 test_PHP 1.0 --- start php seay 2016-09-06 17:17:20
8 test_java_specical 12 --- start Java 长虹 2016-09-05 15:05:06
9 test_PHP 22 --- start php 长虹 2016-09-05 14:16:21
10 test_java_chenyy 1.1 --- start Java test_luo 2016-08-31 11:19:16
11 test_java_chenyy v1.0 --- start Java test_luo 2016-08-31 10:58:20
12 test_PHP 1.2.1 --- start php seay 2016-08-18 15:57:13
13 javaSpacial v1.0 --- start Java admin 2016-08-17 13:04:24
14 webgoat 1.0 --- start Java --- 2016-08-16 21:27:21
15 test_PHP 保险业务 --- start Java xiechungang 2016-08-16 09:39:28
16 test_java_specical v1.0 --- start Java test_luo 2016-08-12 10:08:58
17 Charater-xml 1.0 --- start Java admin 2016-08-11 18:56:09
18 webgoat1.0 ver1.0 --- start Java --- 2016-08-10 16:26:40
19 webgoat1.0 1.0xie --- start Java admin_xie 2016-08-08 11:03:11
20 test_JS v1.1 --- start JS admin 2016-08-03 15:36:55
Page 1 of 1 每页 25 条记录
显示 1 - 25条记录，共 25条记录

找八哥——SVN管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
项目名称：
搜 索
新增JAVA项目
项目名称：
项目版本：
请选择JDK的版本：
项目编码：
上传项目源代码(zip,rar)：
上传jar包(zip,rar,jar)：
从SVN下载源码：
SVN配置
SVN URL：
用户 名：
密 码：
密码确认：
SVN策略：
(小时) 开始时间：
（注：1天=24小时，1周=168小时，1月=720小时）
是否启用计划任务：
是
否
保存配置
删除配置
关 闭
添加项目名称
测试人员
创建时间
webgoat1.0
test_JS
v1.1
admin
2016-08-03 15:36:55
Page 1 of 1
每页 25 条记录
显示 1 - 25条记录，共 25条记录

找八哥---结果管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
项目名称：
搜 索
删除 | 分享结果
项目名称 版本 开发语言 测试状态 查看结果 测试人员 测试时间 查看文件
1 test-net webgoat C#.net 概况 审计 admin 2016-09-30 18:51:02 日志 文件 历史
2 test-net 1.0 C#.net 概况 审计 admin 2016-09-28 18:19:29 日志 文件 历史
3 test-net v1.0 C#.net 概况 审计 admin 2016-09-28 18:18:49 日志 文件 历史
4 test_PHP 1.0 php 概况 审计 seay 2016-09-06 17:17:33 日志 文件 历史
5 test_java_specical 12 Java 概况 审计 长虹 2016-09-05 15:30:53 日志 文件 历史
6 test_PHP 22 php 概况 审计 长虹 2016-09-05 14:24:49 日志 文件 历史
7 test_java_chenyy v1.0 Java 概况 审计 test_luo 2016-08-31 13:31:15 日志 文件 历史
8 test_java_chenyy 1.1 Java 概况 审计 test_luo 2016-08-31 11:19:28 日志 文件 历史
9 javaSpacial v1.0 Java 概况 审计 admin 2016-08-29 23:19:58 日志 文件 历史
10 test_PHP 保险业务 Java 概况 审计 xiechungang 2016-08-20 23:33:02 日志 文件 历史
11 test_java_specical v1.0 Java 概况 审计 test_luo 2016-08-18 15:53:55 日志 文件 历史
12 Charater-xml 1.0 Java 概况 审计 admin 2016-08-11 18:56:13 日志 文件 历史
13 webgoat1.0 1.0xie Java 概况 审计 admin_xie 2016-08-08 11:03:17 日志 文件 历史
14 webgoat 1.0 Java 概况 审计 xiaomeng 2016-08-04 10:44:54 日志 文件 历史
Page 1 of 1 每页 25 条记录
显示 1 - 18条记录，共 18条记录

找八哥---查看结果

文件 视图 模板 格式 报告 高级 帮助

漏洞列表 危险 (117) 搜索

危(117) 高(179) 中(10) 低(970) All(1276)

- 跨站脚本:DOM型 [0/3]
- 跨站脚本:持久式 [0/28]
- 动态代码执行:代码注入 [0/2]
- 密码管理:硬编码密码 [0/16]
- 私密违反 [2/67]
- SQL 注入 [0/1]
  - WsSqlInjection.java:250

漏洞追踪

- WsSqlInjection.java:262 getCreditCard(0)
- WsSqlInjection.java:264 getResults(0)
- WsSqlInjection.java:244 Assignment to query
- WsSqlInjection.java:250 executeQuery(0)

源代码: WsSqlInjection.java

```
236     }
237
238
239     public ResultSet getResults(String id)
240     {
241     try
242     {
243         Connection connection = DatabaseUtilities.getConnection("guest", getWebgoatContext());
244         String query = "SELECT * FROM user_data WHERE userid = " + id;
245         try
246         {
247         Statement statement = connection.createStatement(
248             ResultSet.TYPE_SCROLL_INSENSITIVE,
249             ResultSet.CONCUR_READ_ONLY);
250         ResultSet results = statement.executeQuery(query);
251         return results;
252         }
253         catch (SQLException sqle)
254         {}
255     }
256     catch (Exception e)
257     {}
258     return null;
259     }
260
261
262     public String[] getCreditCard(String id)
```

概要 解释说明 修复建议

解释说明

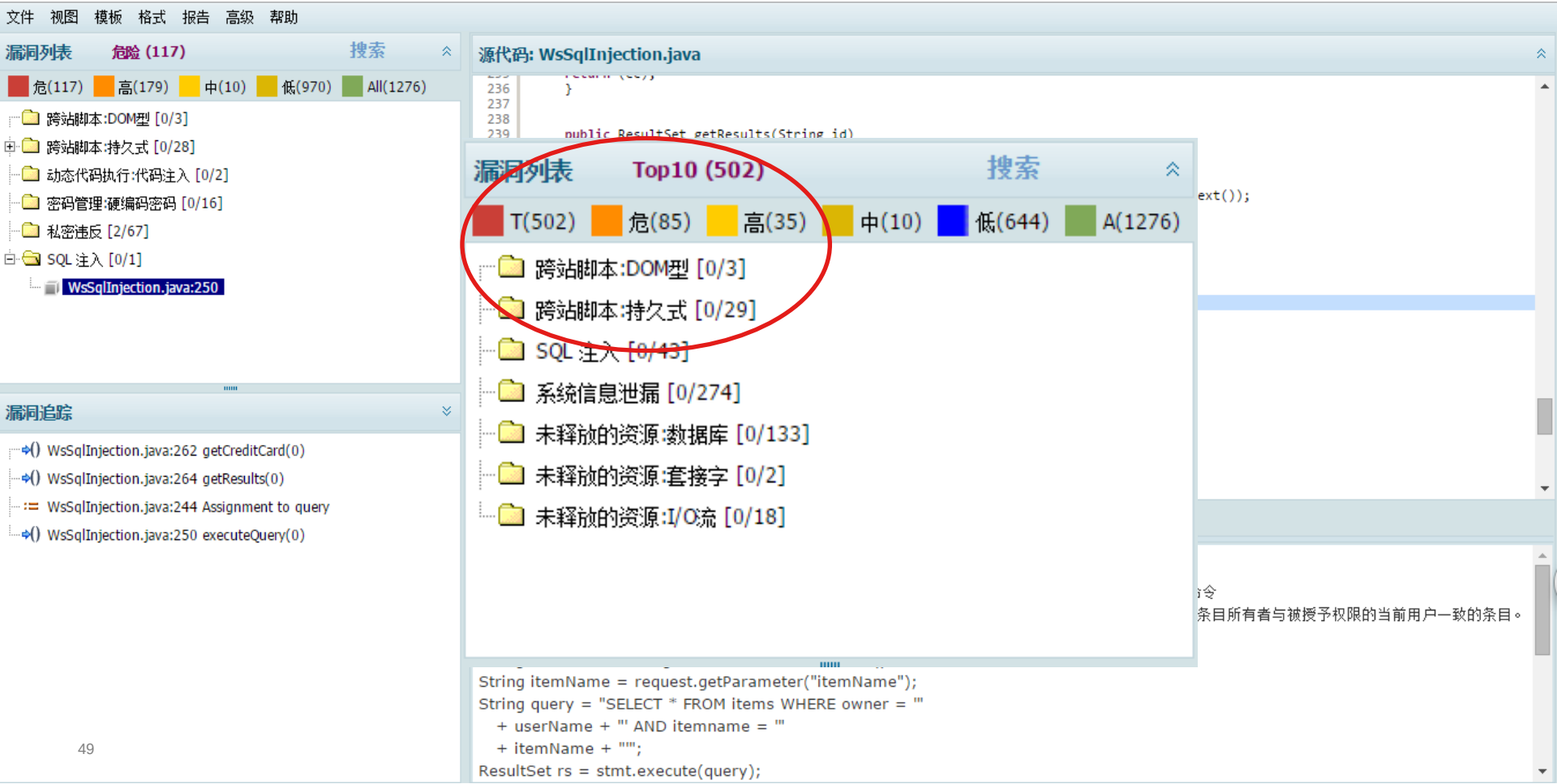
将不可信的输入数据用于构建动态（拼接到） SQL 指令，攻击者就能够修改指令的含义或者执行任意 SQL 命令

例 ： 以下代码动态地构造并执行了一个 SQL 查询，该查询可以搜索与指定名称相匹配的项。该查询仅会显示条目所有者与被授予权限的当前用户一致的条目。

...

```
String userName = ctx.getAuthenticatedUserName();
String itemName = request.getParameter("itemName");
String query = "SELECT * FROM items WHERE owner = '"
  + userName + "' AND itemname = '"
  + itemName + "'";
ResultSet rs = stmt.execute(query);
```

找八哥---自定义安全级别

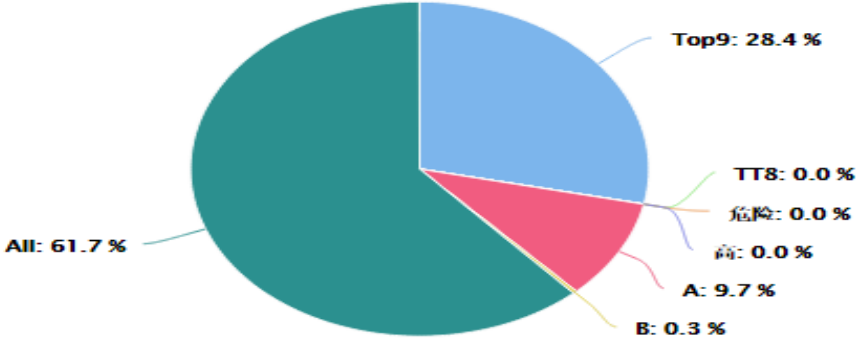

文件 视图 模板 格式 报告 高级 帮助
漏洞列表 危险 (117) 搜索
危(117) 高(179) 中(10) 低(970) All(1276)
跨站脚本:DOM型 [0/3]
跨站脚本:持久式 [0/28]
动态代码执行:代码注入 [0/2]
密码管理:硬编码密码 [0/16]
私密违反 [2/67]
SQL 注入 [0/1]
WsSqlInjection.java:250
漏洞追踪
WsSqlInjection.java:262 getCreditCard(0)
WsSqlInjection.java:264 getResults(0)
WsSqlInjection.java:244 Assignment to query
WsSqlInjection.java:250 executeQuery(0)
源代码: WsSqlInjection.java
236 }
237
238
239 public ResultSet getResults(String id)
漏洞列表 Top10 (502) 搜索
T(502) 危(85) 高(35) 中(10) 低(644) A(1276)
跨站脚本:DOM型 [0/3]
跨站脚本:持久式 [0/29]
SQL 注入 [0/43]
系统信息泄漏 [0/274]
未释放的资源:数据库 [0/133]
未释放的资源:套接字 [0/2]
未释放的资源:I/O流 [0/18]
ext());
令
条目所有者与被授予权限的当前用户一致的条目。
String itemName = request.getParameter("itemName");
String query = "SELECT * FROM items WHERE owner = '"
+ userName + "' AND itemname = '"
+ itemName + "'";
ResultSet rs = stmt.execute(query);

找八哥---自定义报告

一.报告汇总信息

项目名称：test_JAVA	测试人员：ranran
项目版本：v1.2	开发语言：JAVA
扫描文件数(个)：382	代码行数(行)：40489
漏洞数量(个)：1276	有效代码行数(行)：12103
部门：测试部	测试时间：2016年07月31日

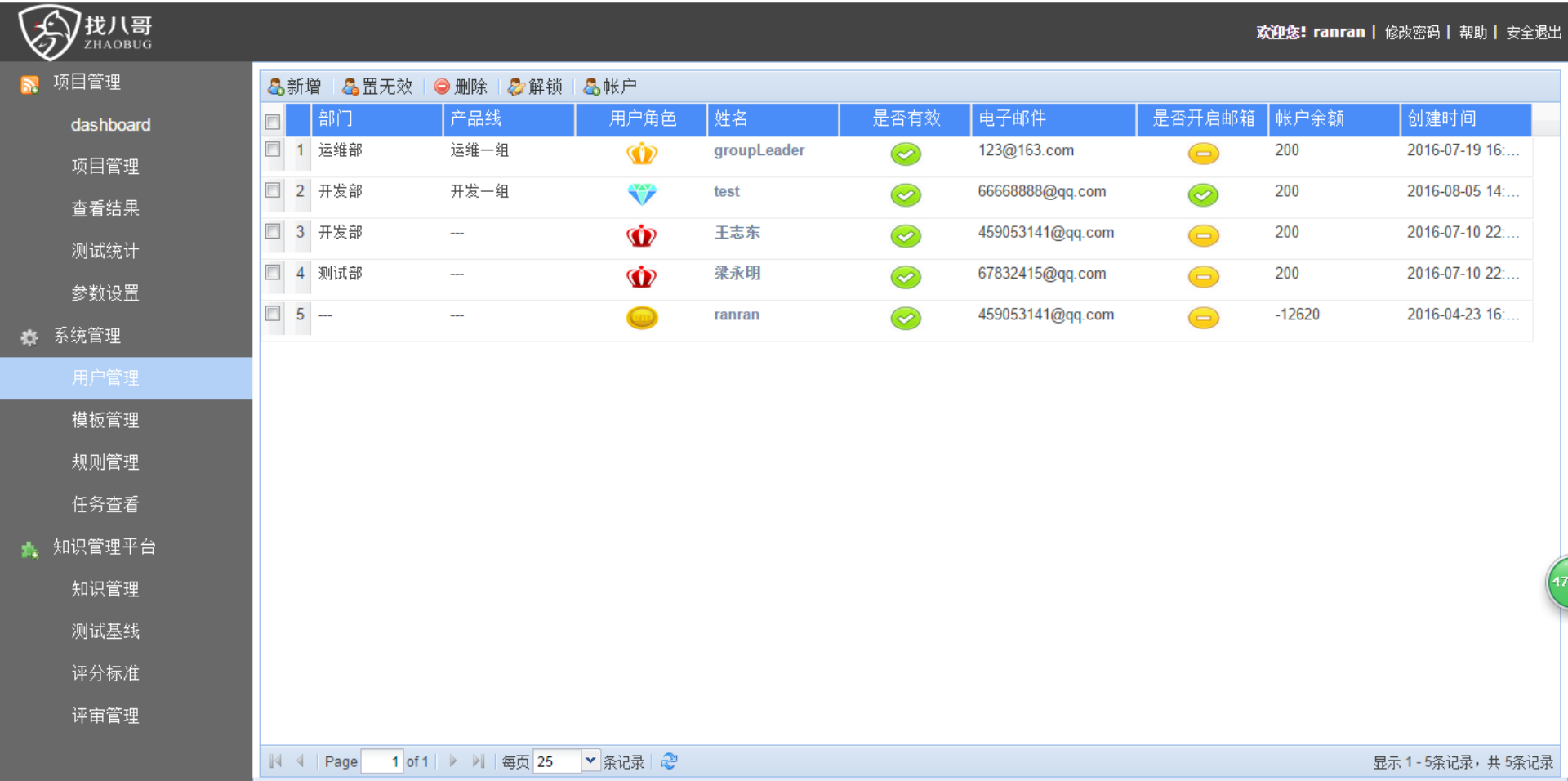
漏洞分布图



找八哥---结果统计


找八哥
ZHAOBUG
欢迎您: ranran | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
维度选择：
部门
产品线
项目
版本
语言
人员
时间选择：
2016-01-01
2016-12-31
查 询
测试总次数分布图
运维一组
运维二组
测试三组
测试一组
测试总次数和成功次数分布图
32
28
24
20
16
12
8
4
0
运维一组
运维二组
测试三组
测试一组
测试总次数
成功次数
部门名称
产品线名称
项目名称
版本
项目语言
测试员
测试总次数
测试成功的...
测试失败的...
花费(元)
1 测试部 测试一组 --- --- --- --- 26 3 23 300
2 开发部 开发一组 --- --- --- --- 1 1 0 100
Page 1 of 1
每页 25 条记录
显示 1 - 2条记录，共 2条记录

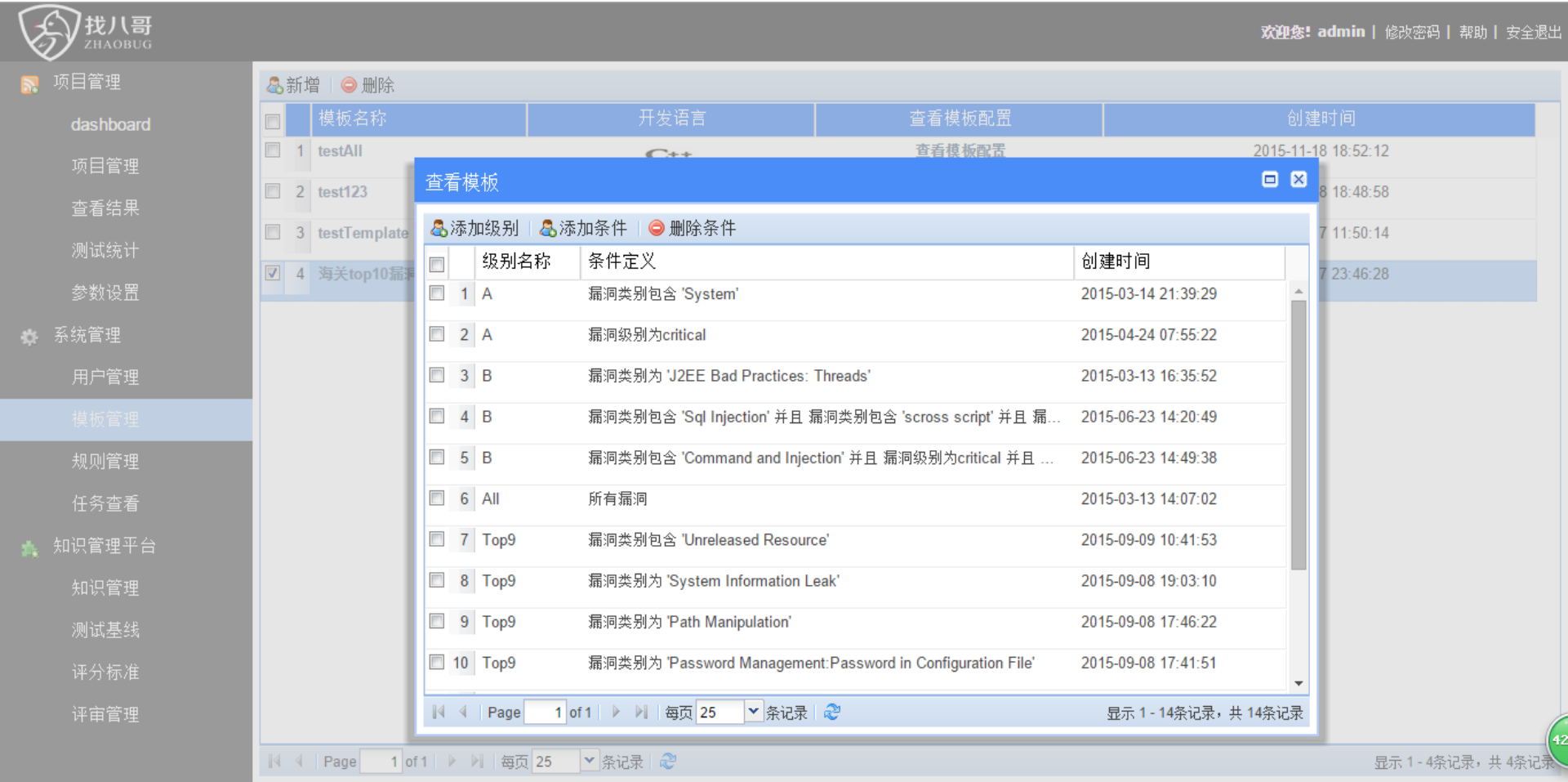
找八哥---用户管理


找八哥
ZHAOBUG
欢迎您! ranran | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
新增 | 置无效 | 删除 | 解锁 | 帐户
部门 产品线 用户角色 姓名 是否有效 电子邮件 是否开启邮箱 帐户余额 创建时间
1 运维部 运维一组 groupLeader 123@163.com 200 2016-07-19 16:...
2 开发部 开发一组 test 66668888@qq.com 200 2016-08-05 14:...
3 开发部 --- 王志东 459053141@qq.com 200 2016-07-10 22:...
4 测试部 --- 梁永明 67832415@qq.com 200 2016-07-10 22:...
5 --- --- ranran 459053141@qq.com -12620 2016-04-23 16:...
Page 1 of 1 每页 25 条记录
显示 1 - 5条记录，共 5条记录

找八哥—参数管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
参数设置
启用64位JDK (推荐启用)
启用debug模式 (推荐启用,日志更详细易于排错)
测试结果不加载源代码方法的详细信息 (推荐启用,测试速度快)
测试过程减少对磁盘的访问 (启用此参数需要足够大内存支持,如4096m)
指定zbg最大使用内存的大小M (如:1024或4096)
4096
指定JVM的Perm最大值 (如:128或512)
516
指定分析JSP的内存最大值 (如:1024)
1024
邮箱设置:
邮件服务器:
smtp.163.com:25
(如:smtp.163.com:25)
邮箱地址:
flying@163.com
邮箱帐号:
flying
邮箱密码:
保 存

找八哥—标准管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
新增
删除
模板名称
开发语言
查看模板配置
创建时间
1 testAll
2 test123
3 testTemplate
4 海关top10漏洞
查看模板配置
2015-11-18 18:52:12
8 18:48:58
7 11:50:14
7 23:46:28
查看模板
添加级别
添加条件
删除条件
级别名称
条件定义
创建时间
1 A 漏洞类别包含 'System' 2015-03-14 21:39:29
2 A 漏洞级别为critical 2015-04-24 07:55:22
3 B 漏洞类别为 'J2EE Bad Practices: Threads' 2015-03-13 16:35:52
4 B 漏洞类别包含 'Sql Injection' 并且 漏洞类别包含 'scross script' 并且 漏... 2015-06-23 14:20:49
5 B 漏洞类别包含 'Command and Injection' 并且 漏洞级别为critical 并且 ... 2015-06-23 14:49:38
6 All 所有漏洞 2015-03-13 14:07:02
7 Top9 漏洞类别包含 'Unreleased Resource' 2015-09-09 10:41:53
8 Top9 漏洞类别为 'System Information Leak' 2015-09-08 19:03:10
9 Top9 漏洞类别为 'Path Manipulation' 2015-09-08 17:46:22
10 Top9 漏洞类别为 'Password Management:Password in Configuration File' 2015-09-08 17:41:51
Page 1 of 1
每页 25 条记录
显示 1 - 14条记录，共 14条记录
Page 1 of 1
每页 25 条记录
显示 1 - 4条记录，共 4条记录

找八哥—规则库管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
正在使用的规则：（当前版本:2016.2.1.0001）
zbg_android.bin
zbg_annotations.bin
zbg_config.bin
zbg_content.bin
zbg_cpp.bin
zbg_dotnet.bin
zbg_ext_cpp.bin
zbg_ext_dotnet.bin
zbg_ext_java.bin
zbg_ext_javascript.bin
zbg_ext_sql.bin
zbg_java.bin
zbg_javascript.bin
zbg_jsp.bin
zbg_objc.bin
zbg_php.bin
zbg_python.bin
zbg_sql.bin
zbg_vb.bin
正在使用的自定义规则：
RCB_ErrorCharacter.xml
删 除
发布新规则：
默认规则库
选择文件
未选择任何文件
自定义规则库
选择文件
未选择任何文件
上传并发布

找八哥—任务管理


找八哥
ZHAOBUG
欢迎您! admin | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
扫描服务器运行情况一览
静态代码扫描服务：
正在运行的任务(0)
正在等待的任务(0)
各主机运行情况：
机器Ip:port
机器是否可用
正在运行的任务数
127.0.0.1:8080
可用
0
当前在线用户：
zhu admin
重置集群

找八哥—评审权重管理


找八哥
ZHAOBUG
欢迎您! ranran | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
项目权重设置
任务名称 分值 部门 产品线
1 cloud-compiler 1 开发部 开发二组
2 cloud-service 1 开发部 开发一组
3 cloud-scaner 1 开发部 开发一组
4 test_ASP 1 测试部 测试一组
5 test_SQL 1 测试部 测试一组
6 test_PHP 5 测试部 测试一组
7 test_net 1 测试部 测试一组
Page 1 of 1 每页 25 条记录 显示 1 - 8条记录，共 8条
级别权重设置
模板名称 级别名称 级别权重
1 系统默认模板 危险 8
2 系统默认模板 高 3
3 系统默认模板 中 1
4 系统默认模板 低 0.01
5 top10漏洞模板 Top10 8
6 top10漏洞模板 危险 1.5
7 top10漏洞模板 高 8
Page 1 of 1 每页 25 条记录 显示 1 - 16条记录，共 16
审计项权重设置
审计项名称 审计项权重
1 未做审计 0.1
2 需要修复的漏洞 13
3 不确定是否为漏洞 3
4 不需要修复的漏洞 0.5
评分标准
序号 评分等级 评分权值
1 A（优）： <100
2 B（良好） >=100 & <200
3 C（一般） >=200 & <500
4 D（差）： >=500

找八哥—安全知识管理

思客云软件源代码安全漏洞知识平台

- 安全漏洞知识库
  - JAVA
    - 输入验证
    - API误用
    - 异常处理
    - 质量性能
    - 代码规范
    - 安全控制
    - 环境配置
    - 信息封装
    - 国内特色
  - HTML
  - .NET
  - Cobol
  - C++
  - JavaScript
  - Objective-C
  - PHP
  - Python
  - XML
  - SQL
  - ASP

空指针引用：Null Dereference

解释说明

程序对一个可能为空指针引用进行操作前未对其进行空指针判断，可能引起 NullPointerException 异常
例：在以下代码中，程序员假定系统始终会定义一个名为"cmd"的属性。如果攻击者能够控制程序环境以使"cmd"变成未定义的属性，那么程序会在尝试调用 trim() 方法时抛出一个 Null 指针异常。

```
String val = null;
...
cmd = System.getProperty("cmd"); if (cmd)  val = util.translateCommand(cmd);
...
cmd = val.trim();
```

修复建议

对任何可能为空指针的引用进行操作前，进行有必要的空指针检查，程序会更加健壮

参考资料

Common Weakness Enumeration - (CWE) CWE ID 476

NIST Special Publication 800-53 Revision 4 - (NIST SP 800-53 Rev.4) SC-5 Denial of Service Protection (P1)

OWASP Top 10 2004 - (OWASP 2004) A9 Application Denial of Service

Payment Card Industry Data Security Standard Version 3.0 - (PCI 3.0) Requirement 6.5.5

Payment Card Industry Data Security Standard Version 3.1 - (PCI 3.1) Requirement 6.5.5

找八哥—审计标准管理

项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理

思客云软件安全测试标准(基线)

软件安全〔2016〕001号

随着互联网环境日益复杂，不安全软件所带来的社会负面效应和经济损失变得触目惊心。如何确保软件安全已经成为我公司信息安全建设进程中的重中之重。为了响应相关部门对信息安全的要求，加强我公司信息安全化建设，增强软件系统抵御攻击的能力，我公司要求所有信息系统进行软件安全性测试。

软件安全涵盖了软件的业务需求、框架设计、开发语言特性等诸多方面。我公司要求从软件的安全测试入手，建立一套切实可行的应用软件源代码安全性测试规范。该规范从输入数据处理、数据库访问、系统资源管理、信息管理等多个方面规定了以JAVA语言开发的WEB应用系统TOP 10种安全隐患和以C/C++语言开发的应用系统的TOP10种安全隐患，以此来区分不同种类、不同性质、不同环境的应用系统之间安全问题的不同特点。根据每种安全隐患可能带来的危害性、可能带来的社会负面效应、可利用性和修复安全隐患的难易程度等因素的综合考虑，进行了如下排名。我公司将检查所有应用系统检查是否存在TOP 10安全漏洞的情况。

JAVA语言WEB应用安全漏洞TOP10 列表

序号	漏洞类别	包括漏洞子类
		SQL Injection

找八哥——评分标准


找八哥
ZHAOBUG
欢迎您! ranran | 修改密码 | 帮助 | 安全退出
项目管理
dashboard
项目管理
查看结果
测试统计
参数设置
系统管理
用户管理
模板管理
规则管理
任务查看
知识管理平台
知识管理
测试基线
评分标准
评审管理
思客云软件源代码安全评分标准
为了更全面、合理的对软件源代码的安全特性进行评估，更统一、直观、量化的反映安全测试结果。思客云根据多年对用户做软件安全评估服务的经验，制定了一套更为合理且行之有效的软件源代码安全评分标准，该标准可以依据不同的项目权重，不同的漏洞级别以及不同的评审结果对评估项目进行综合评分，使得项目安全评估更加合理化，为项目开发过程中的安全特性的持续改进提供基准和依据。另一方面，该标准为用户内部的项目安全性评比，开发团队安全水平评比，开发人员安全技能评比提供技术支撑。
Score = 项目权重 * (∑ (级别权重 * 缺陷数) + ∑ (审计项权重 * 缺陷数))
序号
评分等级
评分权值
1
A（优）：
<100
2
B（良好）
>=100 & <200
3
C（一般）
>=200 & <500
4
D（差）：
>=500

找八哥的价值

90%
- 扩展License成本降低90%
- 一机多用，用户并发，多机集群，私有化云的部署使用模式。

90%
- 测试人力成本降低90%
- “无人值守”安全测试，减少安全测试人员，更不需要外包安全测试。

90%
- 修复漏洞成本降低90%
- 开发者测试，安全测试提前到编码阶段，极早地发现漏洞，修复漏洞。

90%
- 沟通成本降低90%
- WEB查看、审计漏洞，多部门多角色协同工作，沟通极为简单高效。

90%
- 安全管理成本降低90%
- 平台化体系化的管理系统。管理制度工具化、安全测试工作数据化。

还有吗？

有，

为开发者而生！

找八哥—国内“特色”的规则的扩展

- Hard code: ID Number 硬编码身份证号在代码中
- Hard Code: Credit（Banking) Card Number 硬编码信用卡(银行卡)号在代码中
- Hard code: Password at any element of config 硬编码密码在配置文件任意结节
- Bad Practice: Null at right hand 不好的实践：Null 在后面
- Memory Leak: Tuxedo tpalloc() 内存泄漏：Tuxedo tpalloc()
- Back Door : Time bomb 后门：定时炸弹
- Hard Code :File Separator ：硬编码文件分隔符
- MisUsed the Float type: 错误使用Float 类型
- … 共几十种具有中国特色的规则。

- Hard Code: ID Number 硬编码身份证号在代码
- Hard Code: Card Number or Credit Card Number硬编码银行卡或信用卡号在代码
- Hard Code: Password at any element of config file 硬编码密码在配置文件任意结节
- Hard Code :File Separator硬编码文件分隔符
- Performance Issues: try In loop: 性能问题: 在循环体内try语句
- Performance Issues: Dead loop: 性能问题: 死循环
- Performance Issues: use “+”String: 性能问题: 用”+”拼接字符串
- Performance Issues: use “concat()”String: 性能问题: 用concat()拼接字符串
- Performance Issues: New Object In loop: 性能问题: 在循环体内创建对象
- Performance Issues: Call synchronized method In loop: 性能问题: 在循环体内调用synchronized的方法
- Performance Issues：Deep Nested If Statements：性能问题：深度套嵌if语句

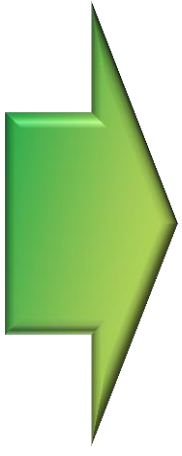
- Performance Issues：Deep Nested Try Catch Blocks性能问题：深度套嵌try catch代码
- Performance Issues: Complex Express In loop: 性能问题: 复杂的表达式在循环体内创建对象
- Performance Issues: Single Character startsWith()：性能问题: 单个字字符用startsWith()查找
- Performance Issues ：No arrange for vector or Hashtable :性能问题: Vector 或 Hashtable 声明时未限定大小
- Performance Issues： Exception as flow control: 性能问题: 用异常控制程序流程
- Performance Issues : Primitive Type Instantiation :性能问题：基础类型不良的实例化方式
- Performance Issues : String Instantiation :性能问题：String类型不良的实例化方式
- Performance Issues : StringBuffer Instantiation With Char 性能问题： StringBuffer实例化单个字符
- Performance Issues : Single Character String 性能问题： 使用单个字符的字符串
- Code Quality：String.toString()：不好的风格：String.toString（）

- Code Quality: Leak return: 代码质量：缺少Return语句
- Code Quality: Null at right hand 代码质量：Null 在后面
- Code Quality: Misused the Float type: 错误使用Float 类型
- Code Quality：Instantiate SimpleDateFormat without Locale: 代码正确性：实例SimpleDateFormat未定locale
- Code Quality: Dobule compare: 代码质量：Double 型比较
- Code Quality: Reassign Parameters : 代码质量: 参数重新赋值
- Code Quality : Class extend Error : 代码质量: 类继承 Erorr 类
- Code Quality : instanceof inside Catch Block 代码质量: instanceof方法在Catch
- Race Condition：Un-synchronized SimpleDateFormat：资源竞争：未同步的simpleDateFormat
- Memory Leak: Tuxedo tpalloc() 内存泄漏：Tuxedo tpalloc()
- Poor Error Handling：Throw NullPointerException：不好的错误处理：抛空指针异常
- J2EE Bad Practices：Forward Inside JSP: J2EE不好的实践：forward在JSP中使用

思客云能帮您带来什么？

---完美的安全审计解决方案和业界最佳的实施经验

让开发者都能爱上安全测试



- 不爱安全测试的程序猿
- 爱上“找BUG(八哥)”的齐天大圣

THANK YOU

王　　宏