1、初识代码审计

1.1、代码审计是什么

代码审计(Code Audit）是一种以发现安全漏洞、程序错误和程序违规为目标的源代码分析技能。在实践过程中，可通过人工审查或者自动化工具的方式，对程序源代码进行检查和分析,发现这些源代码缺陷引发的安全漏洞，并提供代码修订措施和建议。

1.2、代码审计的意义

随着Java Web应用越来越广泛，安全审计已经成为安全测试人员需要直面的工作。虽然PHP在中小型互联网企业仍占据一席之地，但主流的大型应用中，java仍是首选的开发语言，国内外大型企业大多以java作为核心的开发语言。因此对于安全从业者来说，java代码审计已经成为所需要掌握的关键技能。

1.3、代码审计的常用思路

为了在应用代码中寻找目标代码的漏洞,需要有明确的方法论做指导。 方法论 的选择则要视目标程序和要寻找的漏洞类型而定,以下是一些常用思路

(1)接口排查(“正向追踪”):先找出从外部接口接收的参数,并跟踪其传递程,观察是否有参数校验不严的变量传入高危方法中,或者在传递的过程中是否有 代码逻辑漏洞(为了提高排查的全面性,代码审计人员可以向研发人员索要铵以

(2)危险方法溯源(“逆向追踪”)检查敏感方法的参数,并查看参数的传选与 单)。 处理,判断变量是否可控并且已经过严格的过滤。

(3)功能点定向审计:根据经验判断该类应用通常会在哪些功能中出现漏洞

(4)第三方组件、中间件版本比对:检查Web应用所使用的第三方组件或中间 直接审计该类功能的代码。件的版本是否受到已知漏洞的影响。

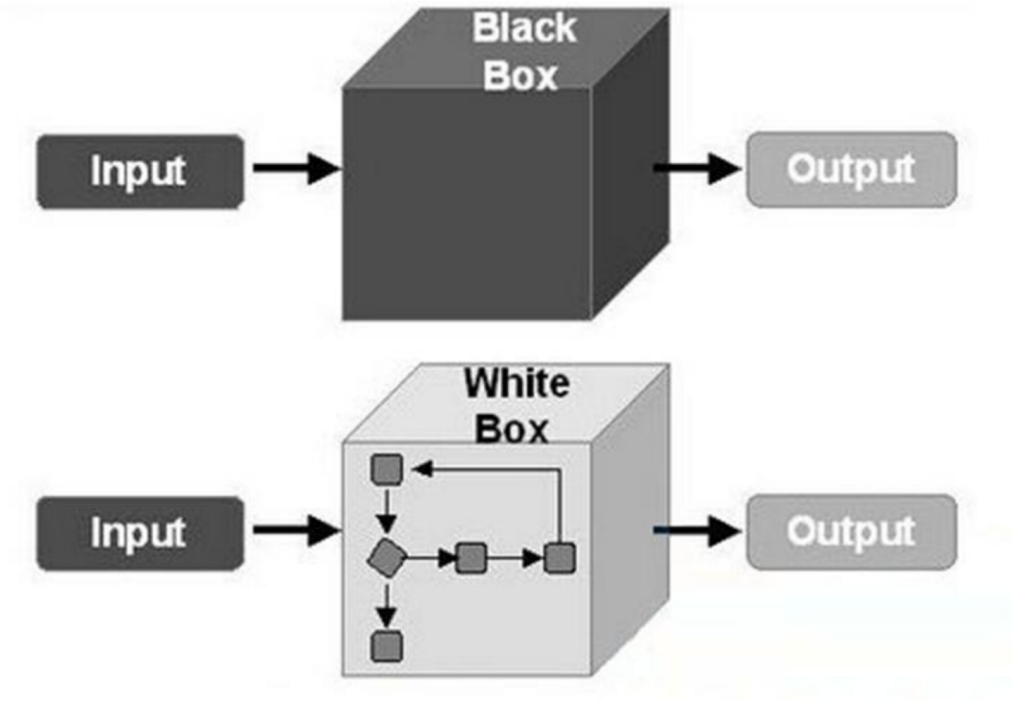
(5)补丁比对:通过对补丁做比对,反推漏洞出处。

(6)“黑盒测试”+“白盒测试”:我认为“白盒测试少直觉,黑盒测试难入做 虽然代码审计的过程须以“白盒测试”为主,但是在过程中辅以“黑盒测试”将有 助于快速定位到接口或做更全面的分析判断。交互式应用安全测试技术IAST就结合 了“黑盒测试”与“白盒测试”的特点。

(7)“代码静态扫描工具”+“人工研判”:对于某些漏洞,使用代码静态扫描工 具代替人工漏洞挖掘可以显著提高审计工作的效率。然而,代码静态扫描工具也存 在“误报率高”等缺陷,具体使用时往往需要进一步研判

(8)开发框架安全审计:审计web应用所使用的开发框架是否存在自身安全 问题,或者由于用户使用不当而引发的安全风险。

1.4 黑盒测试和白盒测试



1.4.1 黑盒测试
黑盒测试又称为功能测试，主要检测软件的每一个功能是否能够正常使用。在测试过程中，将程序看成不能打开的黑盒子，不考虑程序内部结构和特性的基础上通过程序接口进行测试，检查程序功能是否按照设计需求以及说明书的规定能够正常打开使用。

1.4.2 白盒测试
白盒测试也称为结构测试，主要用于检测软件编码过程中的错误。程序员的编程经验、对编程软件的掌握程度、工作状态等因素都会影响到编程质量，导致代码错误

1.5 代码审计和渗透测试的关系

1.5.1 渗透测试优点：

高速提交测试参数，通过前端进行渗透，通过发送数据包到后台服务器

快速发现多层结构的漏洞，根据漏铜的分类，大概能判断是那一层结构的问题。

代码审计优点：

全面，深入的发现漏洞

1.5.2两者之间的关系：

相互补充，彼此强化。
代码审计发现问题，渗透测试确定可利用性。
渗透测试发现问题，代码审计确定成因。

1.6 代码审计的优势

1、提高代码质量

2、降低成本

如果在项目上线后才发现bug 开发人员可能以及调离到其他的项目，协调成本太高，上线后被白帽子发现 成本也很高， 黑客利用 ，数据丢失，成本更高

1.7 代码审计的工作流程